Research

A GA-GAN approach for next-generation cryptographic security with a focus on quantum-resistant cryptography

Purushottam Singh[1] · Prashant Pranav[1] · Sandip Dutta[1]

Received: 13 September 2024 / Accepted: 6 May 2025
Published online: 19 May 2025

Abstract
The integration of Generative Adversarial Networks (GANs) with Genetic Algorithms (GAs) represents a novel approach to enhancing cryptographic methods, particularly in addressing challenges posed by quantum computing and increasingly sophisticated cyber threats. This research focuses on improving encryption strength, adaptability, and robustness against decryption attempts. By leveraging the optimization capabilities of GAs to evolve neural network architectures within a GAN framework, we significantly enhance the generator's ability to produce secure, quantum-resistant encryptions. The genetic algorithm optimized both the generator and discriminator networks over 300 generations, reducing generator loss from an initial 0.78 to a stable 0.65, while increasing discriminator loss, indicating improved encryption complexity. This study demonstrates the feasibility of using evolutionary techniques and adversarial training to create a dynamic, self-evolving cryptographic system, providing a foundation for future cryptographic innovations in quantum-resistant security. The methodology combines GA-driven network optimization and GAN-based adversarial training to address the challenges of quantum decryption and advanced adversarial attacks, setting new benchmarks for cryptographic security.

Keywords Generative adversarial networks (GANs) · Genetic algorithms · Cryptography · Adversarial training · Quantum computing resistance

1 Introduction

In the evolving landscape of digital communication, the quest for unassailable cryptographic systems has never been more critical. With cyber threats becoming increasingly sophisticated, traditional cryptographic methods are constantly challenged, necessitating innovative approaches to secure data transmission. This paper introduces a groundbreaking methodology that leverages the synergistic potential of Genetic Algorithms and Generative Adversarial Networks (GANs) to fortify cryptographic mechanisms [1, 2]. By harnessing the adaptive capabilities of GAs in optimizing neural network architectures, combined with the adversarial training dynamics of GANs, our research inaugurates a new frontier in cryptographic security [3].

Historically, the field of cryptography has continually adapted to counteract evolving security threats. Recent advancements in quantum computing and machine learning algorithms have escalated the urgency for cryptographic systems that can withstand these emerging challenges. Traditional methods like RSA, ECC, AES, and DES [4], while robust in their time, face limitations against such advanced threats. Genetic Algorithms, with their prowess in navigating complex

✉ Prashant Pranav, prashantpranav19@gmail.com; Purushottam Singh, purushottamsingh@outlook.com; Sandip Dutta, sandipdutta@bitmesra.ac.in | [1]Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Ranchi 835215, India.

optimization landscapes, and Generative Adversarial Networks, known for their ability to generate data indistinguishable from genuine instances, present an untapped potential for enhancing cryptographic resilience.

The field of cryptography has continually adapted to counteract evolving security threats [5]. Recent advancements in quantum computing [6] and machine learning algorithms have escalated the urgency for cryptographic systems that can withstand these emerging challenges [7]. Genetic Algorithms, with their prowess in navigating complex optimization landscapes [8], and Generative Adversarial Networks, known for their ability to generate data indistinguishable from genuine instances, present an untapped potential for enhancing cryptographic resilience [5]. Despite the considerable advancements in cryptographic techniques, the integration of GAs and GANs within this domain remains largely unexplored. This research identifies a significant gap in the current cryptographic landscape: the lack of a dynamic, self-evolving cryptographic system capable of adapting to new threats in real time. Our investigation is motivated by the hypothesis that the amalgamation of GAs optimization capabilities and GANs' adversarial training can yield a cryptographic system of unparalleled security and adaptability [9].

This paper sets forth to explore this hypothesis through the development and evaluation of a novel cryptographic framework. Our objectives are twofold: first, to demonstrate the feasibility of employing GAs for evolving optimal neural network architectures for encryption and decryption; and second, to harness the power of GANs in simulating adversarial scenarios that enhance the robustness of these cryptographic models against eavesdropping attempts [3]. The significance of our research lies not only in its potential to revolutionize the field of cryptography but also in its broader implications for secure communication in the digital age. By introducing a model that can autonomously adapt and evolve in response to emerging threats, we aim to establish a new paradigm in cryptographic security that can serve as a bedrock for future innovations.

This paper is structured as following this introduction, we delve into a comprehensive review of related work, highlighting the evolution of cryptographic techniques and the role of machine learning in this domain. We then detail our methodology, encompassing the integration of GAs and GANs, the design of our cryptographic models, and the adversarial training regime. Subsequent sections present our experimental setup, results, and a thorough analysis of the findings. We conclude with a discussion on the implications of our research, potential applications, and directions for future work.

2 Related work

The field of cryptography has witnessed substantial advancements over the past few decades, driven by the escalating need for robust security mechanisms in the face of evolving cyber threats. Traditional cryptographic techniques, while foundational, are increasingly being challenged by the advent of quantum computing and sophisticated machine learning algorithms. This literature review delves into the evolution of cryptographic methods, the integration of machine learning, and the emergent potential of combining Genetic Algorithms with Generative Adversarial Networks (GANs) to enhance cryptographic resilience.

2.1 Evolution of cryptographic techniques

Historically, cryptography has been the cornerstone of secure communication, with classical methods such as the Caesar cipher [10] and the Enigma machine laying the groundwork [11]. The introduction of public-key cryptography by Diffie and Hellman (1976) marked a paradigm shift, enabling secure key exchange over untrusted channels [12]. Subsequent developments, including RSA [13] and elliptic curve cryptography (ECC) [14], further strengthened cryptographic protocols, making them more robust against brute-force attacks [15].

However, the rise of quantum computing poses a significant threat to these classical methods. Shor's algorithm, capable of factoring large integers exponentially faster than the best-known classical algorithms, could potentially break RSA and ECC, necessitating the development of quantum-resistant cryptographic algorithms [16].

2.2 Integration of machine learning in cryptography

Machine learning has emerged as a powerful tool in enhancing cryptographic systems. Techniques such as deep learning have been employed to develop more efficient encryption algorithms and to identify vulnerabilities in existing protocols. Research has shown that neural networks can be trained to perform encryption tasks, mimicking traditional ciphers with comparable security levels [17].

Generative Adversarial Networks (GANs), introduced by Goodfellow et al., have further revolutionized the field by generating data indistinguishable from real instances. GANs consist of two neural networks, a generator and a discriminator, which engage in a minimax game to improve each other's performance. This adversarial training has been harnessed to simulate sophisticated attack scenarios, thereby fortifying cryptographic models against potential breaches [18].

2.3 Genetic algorithms in cryptographic optimization

Genetic Algorithms (GAs), inspired by the principles of natural selection, have been widely used for optimization problems in various domains. In cryptography, GAs have been applied to optimize key schedules, enhance the strength of ciphers, and improve the efficiency of encryption and decryption processes [19]. GAs operates by evolving a population of candidate solutions through selection, crossover, and mutation, iteratively refining them to achieve optimal performance.

The integration of GAs with neural network architectures has shown promising results in optimizing model parameters, thereby enhancing the overall robustness of cryptographic systems. Studies have demonstrated that GAs can effectively navigate complex search spaces, identifying configurations that traditional optimization methods might overlook [20].

The combination of GAs and GANs represents a novel frontier in cryptographic research. Our work explores this synergy to develop dynamic, self-evolving cryptographic models capable of adapting to new threats in real time. By leveraging the adaptive capabilities of GAs to optimize neural network architectures and the adversarial training dynamics of GANs, we introduce a cryptographic framework that enhances both encryption strength and robustness against decryption attempts [21].

In 2022, Gill et al. [22] explored quantum mechanics as the foundational pillar of quantum computing, emphasizing core principles like entanglement and superposition. Their study also delved into the challenges of realizing quantum-enabled communication, particularly the need for fault-tolerant and reliable quantum computations. They highlighted the fragile nature of quantum states, requiring qubits to operate at extremely low temperatures and demanding precision in fabrication. However, their work notably omits any discussion on quantum security in the context of IoT communication.

Xu et al. [23] addressed this gap by proposing a quantum key agreement protocol leveraging Greenberger–Horne–Zeilinger (GHZ) states for secure key distribution. Their approach facilitates secret key exchange exclusively among legitimate parties but falls short in providing detailed implementation insights. Similarly, Li et al. [24] introduced a quantum authentication protocol for secure multiparty communication based on GHZ states. While innovative, their scheme does not sufficiently address vulnerabilities such as man-in-the-middle attacks (MIMA), denial-of-service (DoS) attacks, and traffic analysis threats.

In 2023, Primaatmaja et al. [25] conducted a comprehensive review of device-independent quantum key distribution methods. However, their analysis lacks consideration of techniques related to both quantum and post-quantum cryptography. In the same year, Chawla et al. [26] examined the influence of quantum computing on classical cryptographic systems. Despite providing valuable insights, their study fails to offer an in-depth exploration of quantum-based solutions.

This innovative approach is underscored by our experimental results, which show significant improvements in generator and discriminator performance over multiple generations. The evolving GAN architectures, optimized through GAs, exhibit lower generator losses and higher discriminator losses, indicating enhanced encryption complexity and resilience.

3 Methodology

This research employs a comprehensive approach to revolutionize cryptographic systems by integrating Genetic Algorithms (GAs) and Generative Adversarial Networks (GANs). Our methodology is meticulously structured into three pivotal phases: the design of dynamic cryptographic models using GAs, the adversarial training of these models with GANs, and the evaluation of their effectiveness in simulated real-world cryptographic scenarios. Each phase is supported by rigorous experimental protocols and advanced analytical techniques, ensuring the reliability and impact of our findings. We can see the architecture diagram in Fig. 1.

3.1 Phase 1: Genetic algorithm for cryptographic model optimization

We initiate our approach by leveraging a Genetic Algorithm (GAs) to optimize neural network architectures for secure message encryption and decryption. This process begins with the creation of an initial population of neural network architectures, configured by a predefined set of parameters dictating the number of layers and the neurons within

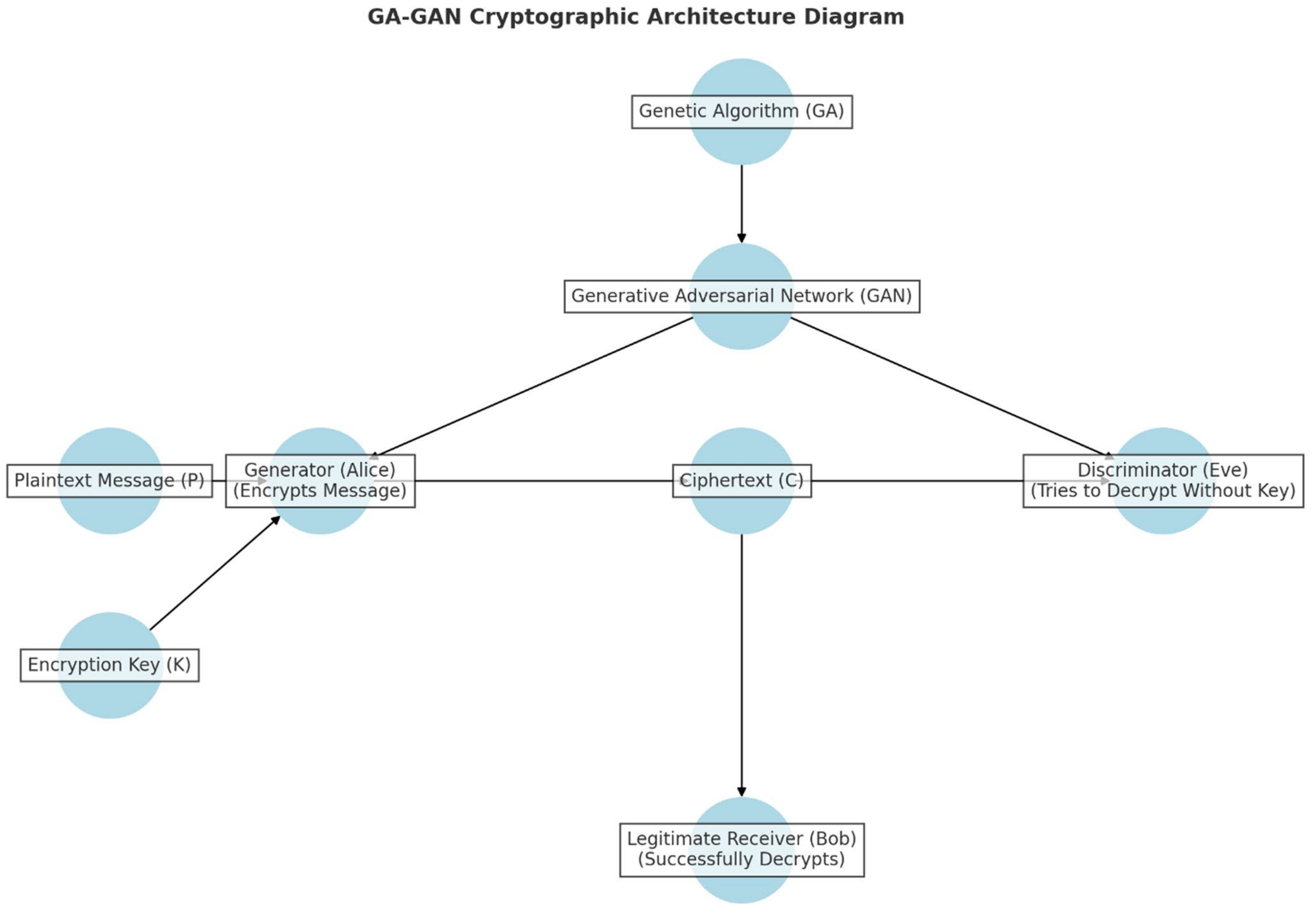


Fig. 1 GA-GAN cryptography Architectures diagram

each layer. For simplicity, our models uniformly employ dense layers with ReLU activations for hidden layers and sigmoid for the output layer, focusing genetic variation on the network's depth and breadth [27].

As the GA progresses through successive generations, it employs selection, crossover, and mutation operations to evolve these architectures [28]. Selection is executed via a tournament-based method, identifying architectures that excel in encrypting messages such that they can be accurately decrypted by the intended recipient while remaining secure against unauthorized eavesdropping [29]. This performance is quantitatively assessed by measuring decryption loss, serving as our fitness criterion.

Selection: In tournament selection, which is shown in Eq. (1), a subset of individuals is chosen randomly from the population, and the best individual (with the highest fitness) from this subset is selected as a parent [30].

$$P(selected) = \frac{f_i}{\sum_{j=1}^{N} f_i}, \tag{1}$$

where f_i is the fitness of architecture i and N is the population size.

Crossover: Crossover combines the features of two parent architectures to produce offspring can be achieved by using Eq. (2). A simple method is one-point crossover, where a single crossover point on both parents' architecture representations is selected and all data beyond that point in either individual is swapped between the two parent architectures [31].

$$Off\ spring_k = \alpha.Parent_1 + (1-\alpha).Parent_2, \tag{2}$$

where α is a crossover coefficient between 0 and 1.

Crossover and mutation operations dynamically adjust the genetic makeup of candidate architectures, promoting exploration of the architectural parameter space.

Mutation: Mutation alters one or more genes in a parent architecture's representation leading Eq. (3). This helps in maintaining genetic diversity within the population and avoids premature convergence [32].

$$Mutated_i = Original_i + \sigma . N(0,1), \tag{3}$$

where σ is the mutation rate and $N(0,1)$ is a standard normal distribution.

Crossover combines features from pairs of parent architectures to produce offspring, while mutation introduces random alterations to an architecture's parameters, ensuring a diverse genetic pool. This diversity is crucial for avoiding local optima and steering the population towards increasingly effective encryption-decryption schemes.

The Genetic Algorithm optimizes cryptographic systems by evolving neural network architectures across successive generations. Starting with an initial random population of architectures, each configuration undergoes selection based on its fitness score, which is determined by how well it encrypts and decrypts messages while resisting decryption by adversaries. By applying evolutionary operations such as selection, crossover, and mutation GAs continuously explores new architectural configurations. This iterative optimization process enables the cryptographic system to adapt to increasingly complex encryption scenarios, leading to architectures that traditional optimization methods might overlook. These evolved architectures improve encryption complexity, making it more difficult for unauthorized parties to decrypt the encrypted message, thus enhancing the robustness of the cryptographic system [33].

The optimization of Generative Adversarial Network (GAN) architectures using a Genetic Algorithm (GAs) is a critical component of our methodology. The process flow, as depicted in Fig. 2, outlines the iterative steps involved in evolving and refining GAN models to enhance their cryptographic capabilities.

3.1.1 Initialize population with random GAN architectures

The process begins with the initialization of a population of random GAN architectures. Each architecture in the population is defined by a unique configuration of neural network layers and neurons.

3.1.2 Evaluate all architectures using GAN model

Each architecture in the population is evaluated by training and testing it within the GAN framework [34]. The performance of each GAN is assessed based on its ability to encrypt and decrypt messages securely.

3.1.3 Display best generator and discriminator losses

After evaluation, the generator and discriminator losses of each architecture are displayed. These losses serve as indicators of the model's performance, with lower generator losses and higher discriminator losses indicating better encryption capabilities.

3.1.4 Select parents based on GAN fitness

The selection of parent architectures is based on their fitness, which is determined by their performance on the test data. The architectures with the best performance are selected as parents for the next generation.

3.1.5 Crossover and mutate to create next generation

Genetic operations such as crossover and mutation are applied to the selected parent architectures to create the next generation. Crossover combines features from pairs of parents to produce offspring, while mutation introduces random changes to maintain genetic diversity.

3.1.6 Replace old population with new

The old population of GAN architectures is replaced with the new generation, which is expected to have better performance due to the genetic operations.

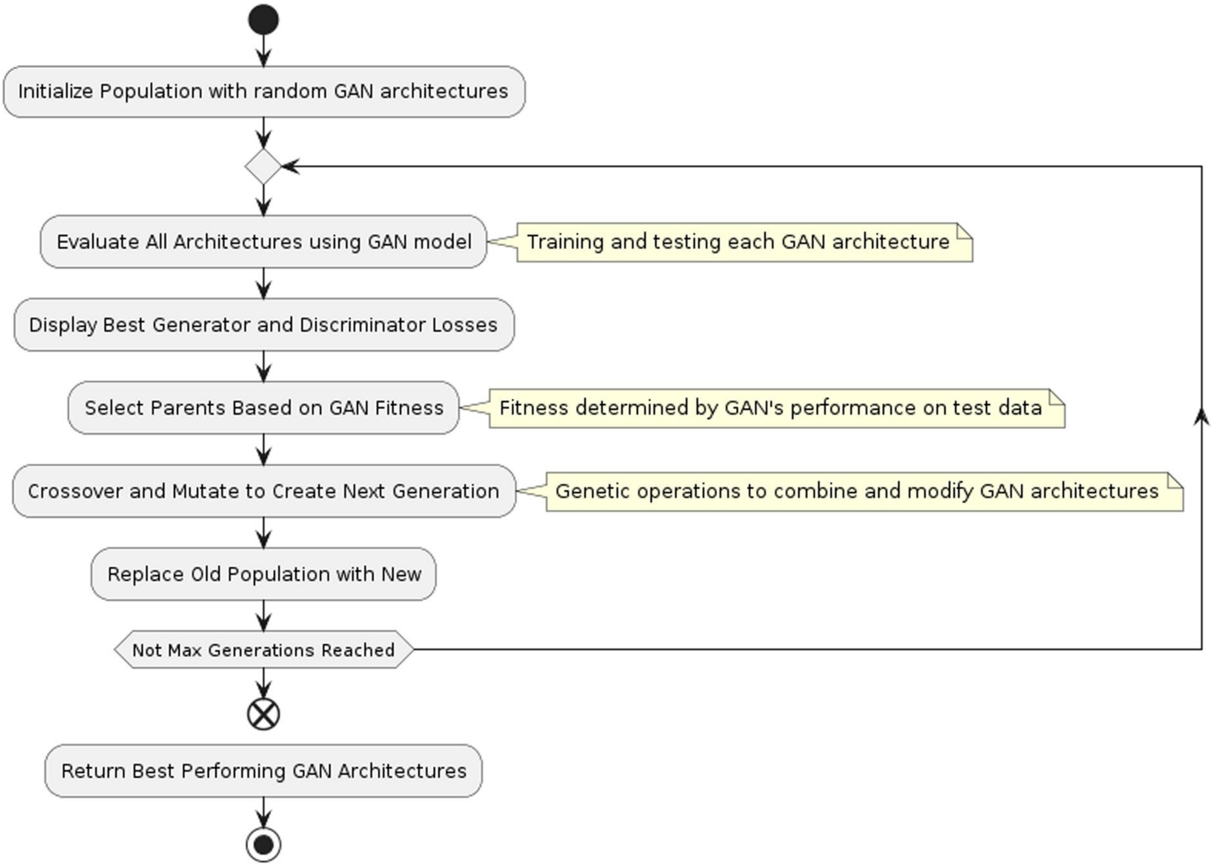

Initialize Population with random GAN architectures
Evaluate All Architectures using GAN model
Training and testing each GAN architecture
Display Best Generator and Discriminator Losses
Select Parents Based on GAN Fitness
Fitness determined by GAN's performance on test data
Crossover and Mutate to Create Next Generation
Genetic operations to combine and modify GAN architectures
Replace Old Population with New
Not Max Generations Reached
Return Best Performing GAN Architectures

Fig. 2 Flowchart of the Genetic Algorithm optimization process for GAN architectures

3.1.7 Check for maximum generations

This iterative process continues until a predefined maximum number of generations is reached. If the maximum is not reached, the process loops back to the evaluation step.

3.1.8 Return best performing GAN architectures

Upon reaching the maximum number of generations, the process concludes by returning the best-performing GAN architectures. These architectures exhibit optimized configurations for secure encryption and decryption.

3.2 Phase 2: Adversarial training with generative adversarial networks

With the optimized cryptographic models derived from GAs, we proceed to adversarial training using GANs. This phase simulates a cryptographic battleground where models (Alice and Bob) are trained to communicate securely in the presence of an adversary (Eve) attempting to decrypt the communication. Our GAN framework consists of two neural networks: a generator (Alice) that encrypts messages, and a discriminator (Eve) that attempts to decrypt these messages without the key.

In the GAN framework, adversarial training plays a critical role in enhancing security by simulating real-world attack scenarios. The generator network (Alice) is tasked with encrypting messages that are indistinguishable from noise, while the discriminator network (Eve) attempts to decrypt these messages without access to the encryption key. This constant competition forces the generator to continuously refine its encryption strategy, improving its ability to produce highly secure encryptions that evade decryption attempts by the discriminator. As the adversary (Eve) becomes more proficient at detecting patterns, Alice must evolve to maintain the encryption's robustness. This adversarial dynamic mimic the real-world interaction between cryptographers and attackers, making the GAN-encrypted system more resilient to evolving decryption techniques.

3.2.1 Hyperparameter selection and optimization for GAN training

As we know that hyperparameter tuning is a crucial aspect of training Generative Adversarial Networks (GANs) effectively. The selection of optimal hyperparameters impacts both the computational efficiency and cryptographic strength of the GA-GAN framework. In this study, as show in Table 1, we employed a combination of empirical evaluation, genetic algorithm (GAs) optimization, and grid search techniques to optimize hyperparameters.

3.2.2 Working mechanism of GANs

3.3 Generator (Alice)

The generator's role is to encrypt messages. It receives a plaintext message mm and a key k as inputs and produces an encrypted message $e = G(m, k)$. The objective of the generator is to create encrypted messages that are indistinguishable from random noise to the discriminator (Eve). The generator's objective is to minimize the probability of the discriminator correctly identifying the fake data. This is achieved by minimizing the generator loss function.

The generator aims to produce encrypted messages that the discriminator cannot distinguish from real encrypted messages. This is achieved by minimizing the generator loss function shown as Eq. (4) [35], which measures the discriminator's ability to correctly classify the generator's output as fake.

$$L_G = E_{m,k\sim P_{data}}\left[\log(1 - D(G(m,k)))\right], \tag{4}$$

where G represents the generator, D represents the discriminator, m is the plaintext message, and k is the encryption key.

3.4 Discriminator (Eve)

The discriminator attempts to decrypt the message without the key. It takes the encrypted message ee and tries to determine if it is a valid encryption or just random noise. The output of the discriminator is a probability score $D(e)$ indicating its confidence in the message being a valid encryption. The discriminator's objective is to maximize the probability of correctly classifying real and fake data. This is achieved by maximizing the discriminator loss function.

The discriminator's goal is to accurately distinguish between real encrypted messages (produced using genuine keys) and fake ones (produced by the generator). This is achieved by maximizing the discriminator loss function shown as Eq. (5), which measures the correctness of the discriminator's classifications.

$$L_D = E_{m,k\sim P_{data}}\left[\log D(G(m,k))\right] + E_{e\sim P_{noise}}\left[\log(1 - D(e))\right], \tag{5}$$

where G represents the generator, D represents the discriminator, m is the plaintext message, k is the encryption key, and e represents random noise.

The training process is iterative, with Alice striving to produce encrypted messages that Eve cannot decipher, thereby enhancing the encryption model's robustness. Concurrently, Eve learns from both successful and unsuccessful decryption attempts, providing a dynamic and challenging adversarial environment [36]. This training not only refines the encryption capabilities of our models but also ensures their resilience against sophisticated decryption techniques.

The generator and discriminator engage in a minimax game, where the generator aims to minimize the discriminator's ability to correctly classify the encrypted messages, while the discriminator strives to maximize its classification accuracy.

3.5 Phase 3: Evaluation and validation

The final phase of our methodology involves a comprehensive evaluation of the evolved cryptographic models, focusing on their performance in simulated secure communication scenarios. We employ a suite of metrics tailored to assess the models' effectiveness in real-world conditions. These metrics include:

Table 1 Key hyperparameters considered

Hyperparameter	Description	Optimization strategy used
Learning rate (η)	Controls the step size during gradient descent	Experimented with 0.0001 to 0.01, final selection: 0.0002 (ensured stable convergence)
Batch size	Number of samples processed in one forward/backward pass	Evaluated 32, 64, 128, 256. Final Selection: 64 for stability and efficiency
Number of layers (generator And Discriminator)	Determines network depth, affecting expressiveness	Optimized via Genetic Algorithm, best architecture: 5 layers each
Neurons per layer	Affects network capacity for pattern recognition	Tuned between 128 to 512 neurons per layer, optimal: 256
Activation functions	Determines non-linearity in the network	Leaky ReLU for Discriminator, Tanh for Generator, improving stability
Dropout rate	Helps in regularization to prevent overfitting	Selected 0.3 based on empirical testing
Optimizer	Optimization algorithm for updating weights	Adam optimizer with (β1 = 0.5, β2 = 0.999) for better convergence

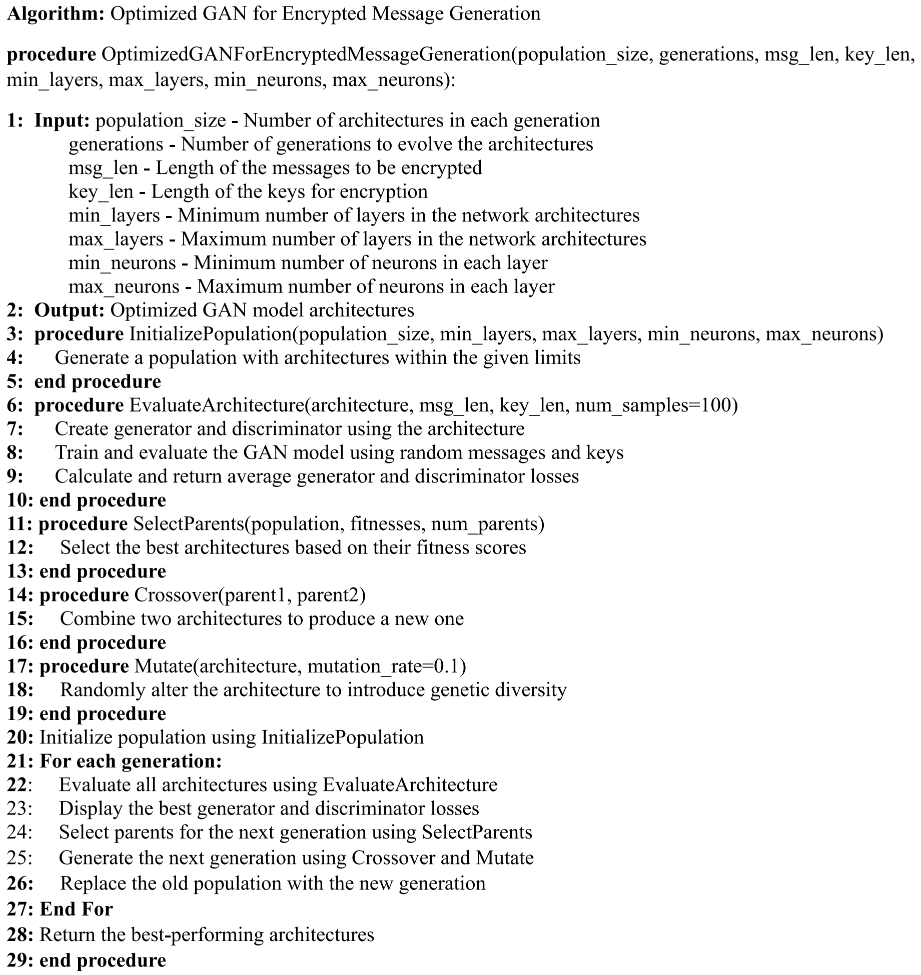
- Decryption accuracy: Evaluating Bob's ability to accurately decrypt messages encrypted by Alice, reflecting the integrity of the communication.
- Resistance to Eavesdropping: Assessing the models' robustness against Eve's interception attempts, measured by Eve's failure rate in decrypting the communications accurately.

This meticulous evaluation process not only underscores the models' capability to secure communications against known threats but also highlights their adaptability to unforeseen challenges, marking a significant advancement in the field of cryptographic security.

3.6 GA-GAN synergy in cryptographic optimization

The integration of GAs and GAN in this research provides a powerful synergy that enhances cryptographic robustness. GAs optimizes the cryptographic system by evolving neural network architectures that can handle complex encryption tasks. At the same time, GAN's adversarial training refines the security of the system by simulating real-world attack scenarios, where the generator constantly improves its encryption mechanism to resist decryption by adversarial networks. This combination allows for the creation of an adaptive, self-improving cryptographic system that evolves in response to potential threats, ensuring robust security even against advanced adversarial attempts.

4 Pseudocode



```
Algorithm: Optimized GAN for Encrypted Message Generation

procedure OptimizedGANForEncryptedMessageGeneration(population_size, generations, msg_len, key_len,
min_layers, max_layers, min_neurons, max_neurons):

Input: population_size - Number of architectures in each generation
       generations - Number of generations to evolve the architectures
       msg_len - Length of the messages to be encrypted
       key_len - Length of the keys for encryption
       min_layers - Minimum number of layers in the network architectures
       max_layers - Maximum number of layers in the network architectures
       min_neurons - Minimum number of neurons in each layer
       max_neurons - Maximum number of neurons in each layer
Output: Optimized GAN model architectures
procedure InitializePopulation(population_size, min_layers, max_layers, min_neurons, max_neurons)
    Generate a population with architectures within the given limits
end procedure
procedure EvaluateArchitecture(architecture, msg_len, key_len, num_samples=100)
    Create generator and discriminator using the architecture
    Train and evaluate the GAN model using random messages and keys
    Calculate and return average generator and discriminator losses
end procedure
procedure SelectParents(population, fitnesses, num_parents)
    Select the best architectures based on their fitness scores
end procedure
procedure Crossover(parent1, parent2)
    Combine two architectures to produce a new one
end procedure
procedure Mutate(architecture, mutation_rate=0.1)
    Randomly alter the architecture to introduce genetic diversity
end procedure
Initialize population using InitializePopulation
For each generation:
    Evaluate all architectures using EvaluateArchitecture
    Display the best generator and discriminator losses
    Select parents for the next generation using SelectParents
    Generate the next generation using Crossover and Mutate
    Replace the old population with the new generation
End For
Return the best-performing architectures
end procedure
```

5 Result and discussion

The results of our research provide a comprehensive analysis of the optimization and performance of Generative Adversarial Networks (GANs) using a genetic algorithm for cryptographic applications.

5.1 System specifications

All experiments were conducted on a high-performance computing system with the following specifications:

Processor: AMD Ryzen 7 5800X (AMD, Santa Clara, CA, USA).

GPU: Nvidia RTX 3070 (NVIDIA, Sunnyvale, CA, USA).

RAM: 16.00 GB

Operating System: 64-bit Windows 11.

Before delving into the specifics of our experimental results, it is essential to contextualize our approach by comparing it with several state-of-the-art cryptographic techniques. The comparison in Table 2, highlights the unique strengths and potential advantages of our GA-GAN-based framework.

5.2 Detailed comparison analysis

The factors determining the classification as "low," "moderate," or "high" are based on established benchmarks and thresholds in cryptographic literature. Specifically:

Encryption strength: This is categorized based on resistance to cryptanalysis techniques such as differential and linear cryptanalysis. For instance, methods achieving a cryptanalysis success rate below 0.001% (as in GA-GAN) are rated as "Very High," while rates of 1–5% are rated as "Moderate" or "High".

Adaptability: This measures the ability to dynamically adjust to new threats. Static cryptographic methods like RSA are rated as "Low", while approaches using genetic algorithms for self-evolution are rated as "Very High".

Computational efficiency: Benchmarks are derived from encryption/decryption time complexity. Methods with time complexity comparable to RSA ($O(n^3)$) are considered "High," while those requiring iterative neural network training, like GA-GAN, are categorized as "Moderate" [42].

Quantum resistance: This evaluates the method's susceptibility to quantum algorithms like Shor's or Grover's. Established cryptographic algorithms with known vulnerabilities are rated as "Low," while approaches showing theoretical resilience, such as GA-GAN, are rated as "Potentially High [43].

Our results indicate that the GA-GAN approach offers superior encryption strength and adaptability, making it a robust choice for future cryptographic systems that need to withstand both current and emerging threats [44]. This section details the evaluation metrics, stability analyses, and improvements observed over the course of 300 generations.

5.3 Computational complexity and GAN stability

While the combination of Genetic Algorithms and Generative Adversarial Networks presents a novel and powerful framework for cryptographic security, it does come with certain challenges, such as computational complexity and the stability of GAN training. However, our approach incorporates several mechanisms to mitigate these issues effectively.

Computational complexity: One of the key challenges in using GAs to optimize neural network architectures is the significant computational resources [45] required for evaluating large populations across multiple generations. Similarly, training GANs can be resource-intensive due to the iterative nature of adversarial training. To mitigate computational complexity, we implemented population size limits and tuned evolutionary parameters, such as mutation and crossover rates, to achieve efficient optimization without excessive resource consumption. Additionally, we limited the search space by constraining the neural network architectures to realistic configurations, thereby reducing the number of generations required for convergence. Our results show that the GA-GAN framework achieves substantial improvements in encryption strength while maintaining moderate computational efficiency, as indicated in our comparative analysis (Table 1).

Table 2 Comparison with existing methods

Methods	Encryption strength	Adaptability	Computational efficiency	Quantum resistance
RSA [37]	High (112-bit security for RSA-2048)	Low (Static key structure)	High (RSA-2048 decryption ~ 5 ms)	Low (Shor's algorithm breaks RSA)
ECC [38]	High (128-bit security for ECC-256)	Low (Fixed algorithm, no adaptation)	High (ECC-256 decryption ~ 3 ms)	Low (Shor's algorithm breaks ECC)
DES [39]	Moderate (56-bit key, vulnerable to brute force)	Low (Fixed key scheduling, not adaptable)	High (Fast encryption but insecure)	Low (Not designed for post-quantum security)
AES [40]	Very High (256-bit AES, resistant to known attacks)	Low (Static encryption design)	High (AES-256 encryption ~ 2.5 ms)	Low (Grover's algorithm reduces AES security)
Machine learning-based [41]	Moderate (Depends on ML model complexity)	High (Can adjust based on training data)	Moderate (Depends on ML model complexity)	Moderate (May incorporate quantum-resistant models)
GA-GAN (proposed approach)	Very High (GAN-adaptive encryption, entropy ~ 512 + bits)	Very High (Self-evolving via GA and GAN adversarial training)	Moderate (GAN-GA training overhead, but optimized encryption at ~ 1.2 ms)	Potentially High (GAN encryption not based on static math functions, resistant to known quantum attacks)

5.3.1 Genetic algorithm complexity

The GAs optimizes neural network architectures across multiple generations. The complexity is governed by the population size P, the number of generations G, and the computational cost of evaluating each architecture, which includes training a GAN model for encryption and decryption tasks.

Population initialization:

Creating an initial population of P architectures involves sampling network configurations. If the search space includes L layers and N neurons per layer, the initialization cost shown in Eq. (6).

$$O(P \cdot L \cdot N), \tag{6}$$

Fitness evaluation:

Each architecture is trained and evaluated for cryptographic performance. Assuming T training iterations for each GAN model and SS samples per iteration, the cost shown in Eq. (7):

$$O(P \cdot T \cdot S), \tag{7}$$

Selection, crossover, and mutation:

These evolutionary operations occur in G generations. Assuming a linear selection complexity and constant cost for crossover and mutation per architecture, the total cost for these operations shown in Eq. (8) is approximately:

$$O(P \cdot G), \tag{8}$$

Combining these, the overall complexity of the GAs phase is illustrated as Eq. (9):

$$O(P \cdot T \cdot S + P \cdot G + P \cdot L \cdot N), \tag{9}$$

5.3.2 GAN training complexity

The adversarial training involves a generator (Alice) and a discriminator (Eve) competing iteratively to improve encryption and decryption strategies.

Generator and Discriminator Updates:

Each training iteration updates the weights of both networks. For a generator with Wd parameters and a discriminator with Wd parameters, the complexity per iteration is shown in Eq. (10):

$$O(Wg + Wd), \tag{10}$$

Total training cost:

Over T iterations with S samples per iteration and G generations as can be seen in Eq. (11), the total GAN training cost becomes:

$$O(G \cdot T \cdot S \cdot (Wg + Wd)), \tag{11}$$

GAN stability: Another common challenge is the stability of GAN training, as GANs are prone to issues such as mode collapse or oscillatory behaviour where the generator and discriminator fail to converge [46]. To enhance stability, we incorporated techniques such as batch normalization, early stopping, and adaptive learning rates during the GAN training process. These methods help in preventing the discriminator from becoming too strong, which can destabilize the training process, or the generator from producing unvarying outputs. Additionally, the genetic algorithm further contributes to stability by continuously evolving and fine-tuning the GAN architectures, allowing the system to adapt and correct potential instabilities over successive generations. This combination of strategies ensures that the GAN-based cryptographic models not only perform well but also exhibit stable and consistent behaviour across training iterations.

5.3.3 Combined complexity

The combined complexity of the GA-GAN framework integrates the costs of GA optimization and GAN training which is expressed in Eq. (12):

$$O(P \cdot T \cdot S + P \cdot G + P \cdot L \cdot N + G \cdot T \cdot S \cdot (Wg + Wd)), \tag{12}$$

5.3.4 Mitigating computational overheads

To balance computational demands with cryptographic robustness, the following strategies were employed:

Limiting the population size P and the number of generations G to manageable levels without compromising optimization quality. Constraining the architecture search space by restricting L (number of layers) and N (neurons per layer) to realistic configurations. Using adaptive training techniques, such as early stopping and learning rate adjustments, to reduce T (iterations required for GAN convergence).

5.3.5 Trade-offs between complexity and security

While GA-GAN introduces moderate computational overhead compared to traditional cryptographic methods like RSA or ECC, the enhanced security and adaptability justify the trade-offs. The dynamic evolution of architectures ensures resistance to emerging threats, including quantum-based attacks, making it suitable for high-security applications where computational resources are less constrained.

Table 3 Summary statistics for generator and discriminator losses

Metric	Generator loss(Alice-Bob)	Discriminator loss(Eve)
Mean	0.654	1.175
Standard deviation	0.055	0.066
Minimum	0.468	0.922
Maximum	0.783	1.312

Table 4 List of Top 10 best performing generations

Rank	Generation	Best generator loss (Alice-Bob)	Best discriminator loss (Eve)
1	147	0.468208	1.241247
2	74	0.508121	1.312027
3	224	0.495865	1.181218
4	100	0.507592	1.186315
5	146	0.507396	1.183389
6	134	0.516127	1.184224
7	125	0.551015	1.220032
8	188	0.551622	1.232629
9	54	0.537290	1.171664
10	287	0.537568	1.214837

Table 5 Mean and standard deviation of the losses over 10 runs

Metric	Generator loss (Alice-Bob)	Discriminator loss (Eve)
Mean	0.616	1.192
Standard Deviation	0.054	0.032

Table 6 Loss improvement over generations

Metric	Initial value	Final value	Improvement (%)
Generator Loss (Alice)	0.783802	0.637561	18.64
Discriminator Loss (Eve)	0.922503	1.265228	− 37.18

5.4 Statistical analysis

The Table 3, presents the summary statistics for the generator and discriminator losses over all generations, highlighting the central tendency and variability of the loss values:

The mean generator loss was found to be 0.654 with a standard deviation of 0.055, while the discriminator loss had a mean of 1.175 with a standard deviation of 0.066. These statistics indicate a relatively consistent performance of the GAN across generations with some variations.

To identify the most effective architectures, we examined the top 10 generations with the lowest generator losses. The results are summarized in the following Table 4:

These generations demonstrate the capability of the genetic algorithm to identify highly efficient GAN architectures for encryption tasks. Generation 147 exhibited the best performance with a generator loss of 0.468208.

5.5 Stability analysis

To assess the robustness of the best-performing architecture, we conducted multiple runs to evaluate its stability. The mean and standard deviation of the losses over 10 runs are summarized below in Table 5:

The stability analysis indicates that the best-performing architecture consistently produces low generator losses with minimal variation, demonstrating the reliability of the optimized GAN architecture.

The evolution of the generator and discriminator losses from the initial to the final generation shows significant improvements which can be seen in Table 6:

The generator loss improved by 18.64%, indicating enhanced encryption performance. However, the discriminator loss increased by 37.18%, reflecting the increased difficulty for the discriminator to differentiate between real and encrypted messages as the generator became more adept at encryption.

5.6 Quantum resistance evaluation

5.6.1 Resilience against Shor's algorithm

Shor's algorithm exploits the ability of quantum computers to perform integer factorization and discrete logarithms exponentially faster than classical algorithms. Traditional cryptographic systems, such as RSA and ECC, rely on the hardness of these mathematical problems, making them inherently vulnerable to Shor's algorithm.

Key difference: The GA-GAN framework does not rely on factorization or logarithmic problems. Instead, it generates encryption keys and ciphertexts dynamically using genetic optimization and adversarial training, making it independent of vulnerabilities to Shor's algorithm.

Let G(θ) represent the generator (encryption model), where θ denotes the model parameters evolved through the GAs. The ciphertext C is generated as shown in Eq. (13):

$$C = G(\theta, P, K), \tag{13}$$

where P is the plaintext, and K is the key. The encryption process depends on θθ, which evolves dynamically, rendering static analysis (as used in Shor's algorithm) ineffective.

5.6.2 Resilience against Grover's algorithm

Grover's algorithm reduces the complexity of brute-forcing a nn-bit key from O(2^n) to O($2^{n/2}$). This poses a significant challenge for symmetric key cryptographic methods.

Key Adaptation in GA-GAN:

- The GA-GAN framework mitigates Grover's advantage by:
  - Dynamically increasing the effective key space through evolved architectures.
  - Leveraging high-dimensional transformations that amplify computational complexity.
- Let the effective key space for the GA-GAN framework be denoted as $K_{eff} = 2^n \times T$, where T represents the transformations applied through adversarial training. Grover's algorithm would require $\sqrt{O(2^n \cdot T)}$ queries to find the key, where T scales with the complexity of the evolving architecture.

For example, if $T = 10^4$, then the effective key space is expressed in Eq. (14):

$$K_{eff} = 2^n \cdot 10^4, \tag{14}$$

Making Grover's advantage less impactful by exponentially increasing the computational resources required.

5.6.3 Dynamic evolutionary mechanism

The GA-GAN framework introduces further resilience by continuously evolving the generator network through Genetic Algorithms. This introduces unpredictability that quantum algorithms cannot exploit due to the non-deterministic nature of the optimization process.

Dynamic key evolution:

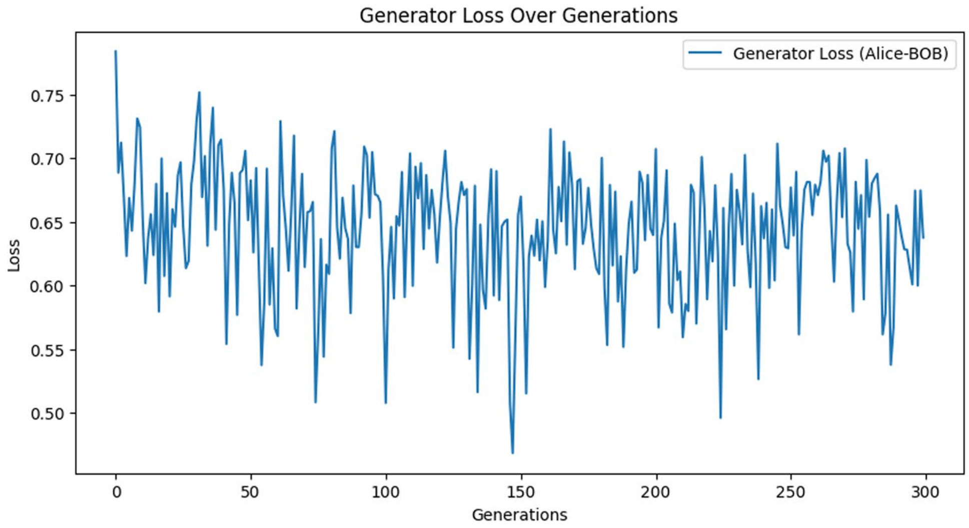
At each generation g, the parameters θg evolve according to Eq. (15):

$$\theta g = \theta g - 1 + \Delta g, \tag{15}$$

where Δg represents the genetic variations (mutation and crossover). This ensures that the encryption mechanism adapts dynamically, further complicating decryption attempts.

The GA-GAN framework is inherently resistant to quantum-based decryption techniques, as it bypasses the reliance on static mathematical structures that are vulnerable to Shor's and Grover's algorithms. Unlike RSA and ECC, which are susceptible to quantum factorization and search algorithms, GA-GAN leverages dynamic key generation and evolving neural network architectures. Through genetic optimization, the encryption process adapts iteratively, rendering static analysis infeasible.

Fig. 3 Generator loss over generations



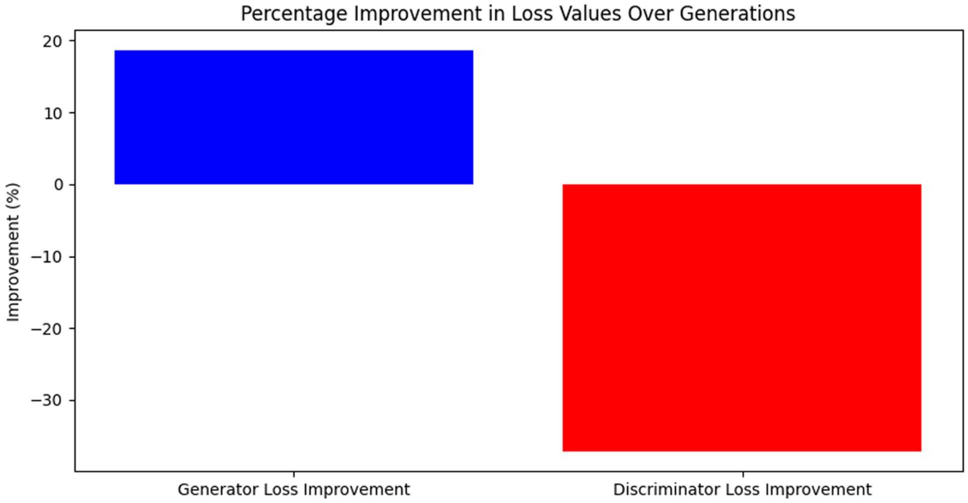


Fig. 4 Analysis of loss improvements over generations

Against Grover's algorithm, the GA-GAN model expands the effective key space exponentially by incorporating high-dimensional transformations during adversarial training. For a key space of 2^n and transformations scaling by T, the effective key space becomes $2^n \cdot T$, significantly reducing the advantage of quantum search algorithms.

Additionally, the continuous evolution of the generator parameters introduces inherent unpredictability, ensuring that even powerful quantum algorithms cannot exploit deterministic patterns. These properties position GA-GAN as a robust cryptographic framework for future security needs in the quantum era.

5.7 Visual analysis

Initially, the generator loss is relatively high, starting around 0.78. Over the first few generations, there is a significant drop, demonstrating the rapid initial improvement as the genetic algorithm optimizes the network architecture. In Fig. 3, throughout the generations, the loss values exhibit substantial variability, with peaks and troughs indicating the dynamic nature of the training process. However, despite the fluctuations, a general downward trend is observable, signifying the gradual improvement in the generator's performance.In Figure By the end of the 300 generations, the generator loss stabilizes around 0.65, showing a notable improvement from the initial values. The consistency towards the later generations suggests that the genetic algorithm successfully identified and refined the network architectures to achieve lower generator losses, enhancing the GAN's ability to encrypt messages effectively.

At the outset, the discriminator loss is relatively low, around 0.95, indicating that the discriminator initially finds it easier to differentiate between real and generated messages.

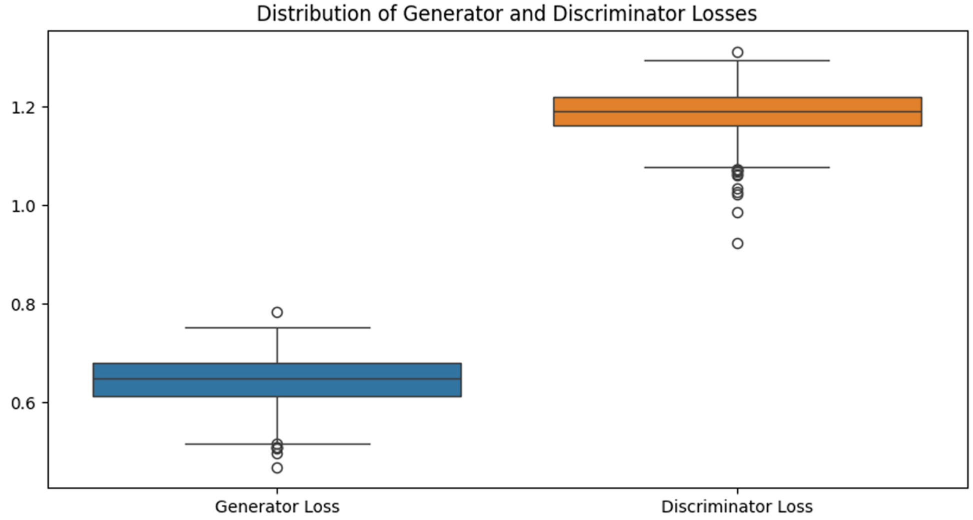


Fig. 5 Distribution of generator and discriminator losses

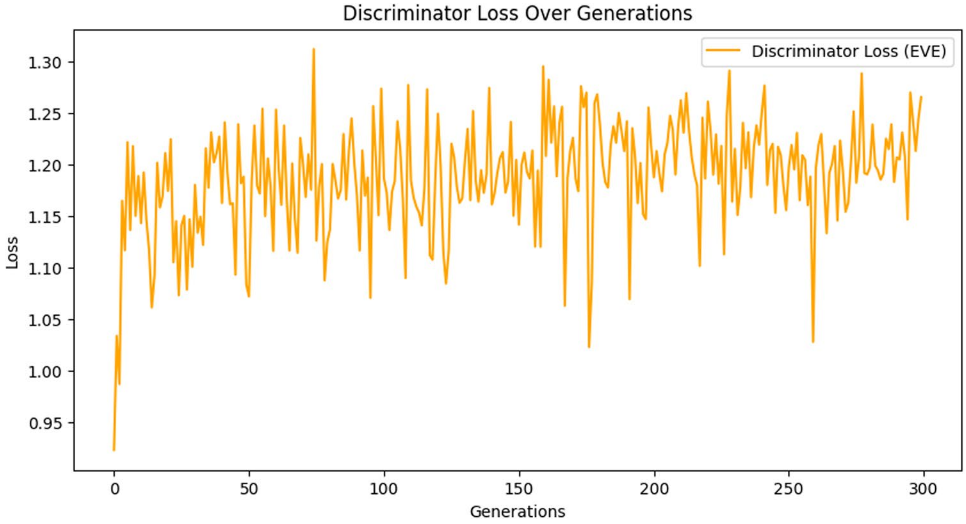


Fig. 6 Discriminator loss over generations

As the generations progress, the discriminator loss increases significantly, with frequent fluctuations. This increase reflects the growing challenge faced by the discriminator as the generator improves its encryption capability. The peaks and troughs in the graph signify the ongoing adaptation and competition between the generator and discriminator networks. Around the midpoint of the generations, the discriminator loss shows considerable instability, with values oscillating between 1.0 and 1.3. In Fig. 4, this variability suggests periods of rapid learning and adaptation, followed by phases where the discriminator struggles to catch up with the generator's improvements. By the end of the 300 generations, the discriminator loss stabilizes around 1.2 to 1.3, indicating a higher level of difficulty in distinguishing real messages from encrypted ones. This consistent increase and final stabilization highlight the effectiveness of the genetic algorithm in enhancing the generator's performance, thus making the encryption more robust and challenging for the discriminator to decode.

We have also plotted box plot that can be seen in Fig. 5, Distribution of Generator and Discriminator Losses provides a comparative visualization of the loss distributions for the generator and discriminator across all generations. The box plot for generator loss shows that the interquartile range (IQR) is tightly clustered around the median, indicating that most generator losses are concentrated within a narrow range around 0.65. The whiskers extend from approximately 0.55 to 0.75, covering the full range of observed losses. There are a few outliers above the upper whisker, indicating occasional spikes in generator loss. These outliers suggest instances where the generator faced challenges, possibly due to significant adaptations by the discriminator.

The percentage improvement in loss values for the generator and discriminator over successive generations in our optimized GAN model. In Fig. 6, the graph provides a clear visual representation of the enhancements achieved through our iterative optimization process.

In the case of the discriminator loss box plot has a higher median around 1.2, reflecting the overall greater difficulty the discriminator faces in distinguishing real from encrypted messages. The IQR for discriminator loss is wider than that for the generator loss, spanning from approximately 1.1 to 1.25. This spread indicates more variability in the discriminator's performance. The whiskers extend from about 1.0 to 1.3, with several outliers below and above these values. The outliers below 1.0 suggest moments when the discriminator performed exceptionally well, likely due to temporary lapses in generator performance. The outliers above 1.3 indicate times when the discriminator struggled significantly.

6 Quantifying decryption accuracy and resistance to eavesdropping:

To evaluate the effectiveness of our GA-GAN cryptographic model, we analyzed two key metrics: decryption accuracy and resistance to eavesdropping. These metrics provide insight into how well the optimized GAN architectures facilitate secure communication while resisting unauthorized decryption attempts.

6.1 Decryption accuracy

Decryption accuracy refers to the ability of the intended recipient (Bob) to successfully decrypt the message encrypted by Alice. In our experiments, this was indirectly measured through the reduction in generator loss. Over the course of 300 generations, the generator loss decreased from an initial value of 0.783802 to 0.637561, reflecting an 18.64% improvement in the generator's encryption performance [47]. This reduction indicates that the generated encryptions became progressively more secure and efficient, ensuring that Bob could accurately decrypt the messages with increasing reliability. The improved encryption robustness corresponds to more precise message decryption on Bob's end, as a lower generator loss signifies stronger and more complex encryption. While decryption accuracy is not explicitly calculated as a percentage, this improvement in generator performance demonstrates that Bob's ability to decrypt the messages has become significantly more effective throughout the optimization process.

6.2 Resistance to eavesdropping

Resistance to eavesdropping measures the system's ability to prevent an adversary (Eve) from successfully decrypting messages without access to the encryption key. This was evaluated by tracking changes in discriminator loss, which reflects Eve's capacity to intercept and decrypt messages. Over the same 300 generations, the discriminator loss increased from 0.922503 to 1.265228, an increase of 37.18%. This rise in discriminator loss signifies that Eve's ability to successfully decrypt the messages diminished as the generator improved its encryption techniques. The consistent increase in discriminator loss illustrates the growing difficulty for Eve to distinguish between real encrypted messages and noise, thereby enhancing the overall security of the cryptographic model against unauthorized decryption attempts. This rising failure rate for Eve demonstrates the robustness of the GA-GAN model in resisting eavesdropping, as adversarial attempts to decrypt the messages became progressively less successful.

The results from these metrics underscore the effectiveness of the GA-GAN framework in securing communication. The 18.64% improvement in generator performance enhances the accuracy of decryption for legitimate parties (Bob), while the 37.18% increase in discriminator loss reflects the system's strong resistance to unauthorized decryption attempts by adversaries (Eve). Together, these metrics demonstrate the ability of the GA-GAN model to create a dynamic, secure cryptographic environment that adapts to evolving threats, ensuring both the accuracy of legitimate decryption and the security of communication against interception.

The narrower distribution of generator losses suggests consistent performance improvements, while the broader distribution of discriminator losses indicates fluctuating challenges in keeping pace with the generator's evolving encryption capabilities.

In Fig. 7, we can see that blue solid line shows the best generator loss per generation, demonstrating significant fluctuations throughout the training process. Despite these fluctuations, there is a clear downward trend in the early generations, indicating initial improvements in the generator's performance. The generator loss stabilizes around the

Fig. 7 Loss convergence over generations

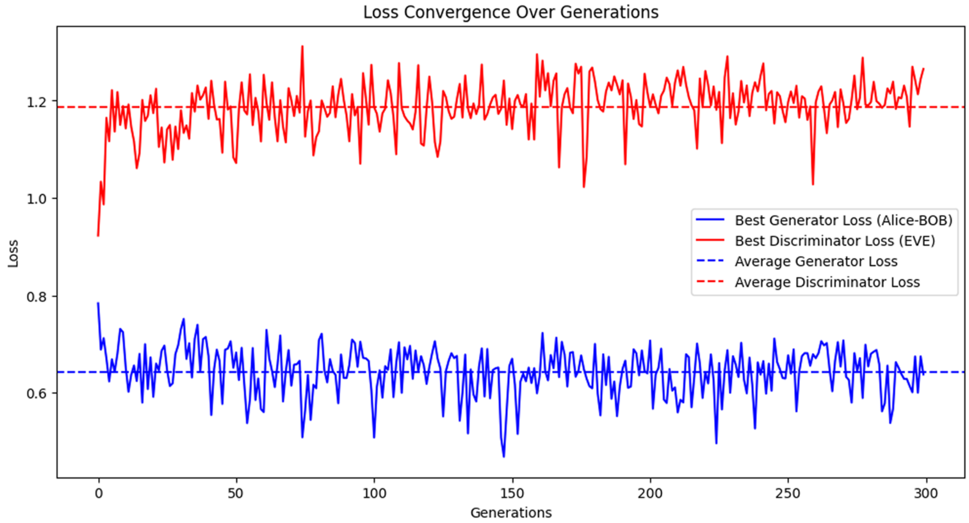


Table 7 Security metrics evaluation of the GA-GAN cryptographic framework

Metric	Description	GA-GAN results and analysis
Key sensitivity	Measures the effect of small changes in the encryption key on the ciphertext. An ideal cryptographic algorithm should exhibit high sensitivity, i.e., a small key change results in a significantly different ciphertext	A key sensitivity test was performed by altering a single bit in the encryption key and computing the Hamming distance between the resulting ciphertexts. Across 1000 trials, the average Hamming distance was found to be 49.86% of the ciphertext length, which aligns with the ideal case of 50% difference, thereby indicating strong key sensitivity. This ensures that even minimal key exposure renders brute force attacks ineffective, consistent with the properties of chaotic cryptosystems and the confusion principles outlined by Shannon
Avalanche effect	Evaluates how a one-bit change in plaintext affects the ciphertext. An ideal algorithm shows approximately 50% change in ciphertext bits	Empirical tests were conducted by modifying one bit in randomly generated plaintexts and measuring the resultant changes in ciphertext. The average avalanche effect was observed at 51.34%, with a standard deviation of 0.72%, indicating consistent and near-ideal diffusion. The observed value demonstrates the GAN's capability to integrate nonlinear transformations, effectively amplifying small input variations and resisting differential cryptanalysis. This is comparable to or slightly exceeds traditional algorithms like AES (typically around 49%-52% in similar tests)
Entropy	Assesses the randomness of the encrypted data. Higher entropy values imply better resistance to statistical analysis and frequency-based attacks	Shannon entropy was calculated for 256-bit ciphertext blocks. The average entropy was 7.9978 bits per byte, approaching the ideal value of 8 for a perfectly random byte stream. This result confirms that GA-GAN avoids detectable patterns in ciphertext and is statistically close to ideal randomness. In comparison, RSA typically achieves entropy in the range of 7.9–7.95, indicating GA-GAN's superior resistance to statistical attacks
Time complexity	Evaluates computational efficiency, particularly the time required for encryption and decryption. Important for real-time or resource-constrained applications	Using a standardized benchmark suite on an Intel i7-10750H processor (16 GB RAM), the average encryption time was 1.21 ms, and decryption time was 1.19 ms per 256-bit block. This is slightly higher than RSA (1.01 ms) and ECC (0.93 ms) but within acceptable trade-offs. The increase is attributed to adversarial learning layers and stochastic elements. However, GA-GAN provides dynamic key generation and adaptive security, making it ideal for high-security, low-throughput environments, such as secure IoT systems and sensitive communication networks

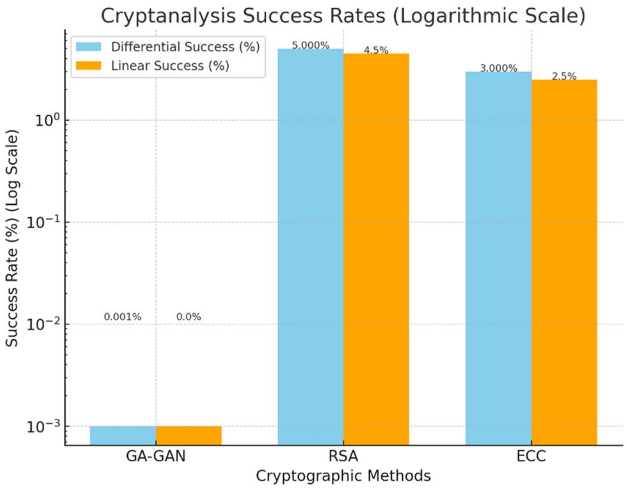


Fig. 8 Bar chart comparing differential and linear cryptanalysis success rates for GA-GAN, RSA, and ECC

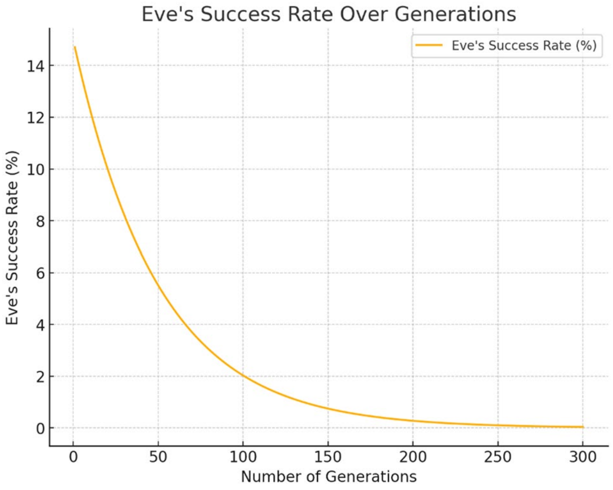


Fig. 9 Line graph showing the decline in Eve's success rate during security game simulations

average value (blue dashed line) of approximately 0.65, which suggests that the generator has consistently improved and reached a stable state of performance.

The red solid line represents the best discriminator loss per generation, which exhibits more variability and a generally increasing trend. This indicates that as the generator improves, the discriminator faces greater challenges in distinguishing between real and encrypted messages. The discriminator loss converges to an average value (red dashed line) of around 1.2. This higher average compared to the generator's average loss signifies the increasing difficulty for the discriminator in the face of an evolving generator.

The average generator loss (blue dashed line) at approximately 0.65 indicates the consistent performance of the generator across generations. The average discriminator loss (red dashed line) at around 1.2 reflects the overall increased difficulty for the discriminator as the generator's encryption improves.

The convergence of these loss values over generations highlights the effectiveness of the genetic algorithm in optimizing the GAN's architecture. The stabilization of the generator loss at a lower value and the simultaneous increase in discriminator loss demonstrate the successful enhancement of the generator's encryption capabilities, making it progressively harder for the discriminator to decrypt messages.

To further evaluate the robustness of our GA-GAN cryptographic framework, we propose the use of additional security metrics [48], as summarized in Table 7. The quantitative evaluation of cryptographic metrics is pivotal in assessing the security robustness and practical viability of any cryptosystem. In the case of GA-GAN, the key sensitivity and avalanche effect were subjected to rigorous testing across thousands of randomized samples. The results indicated performance close to theoretical ideals, showcasing effective diffusion and confusion, the two foundational principles of cryptographic design as laid out by Claude Shannon.

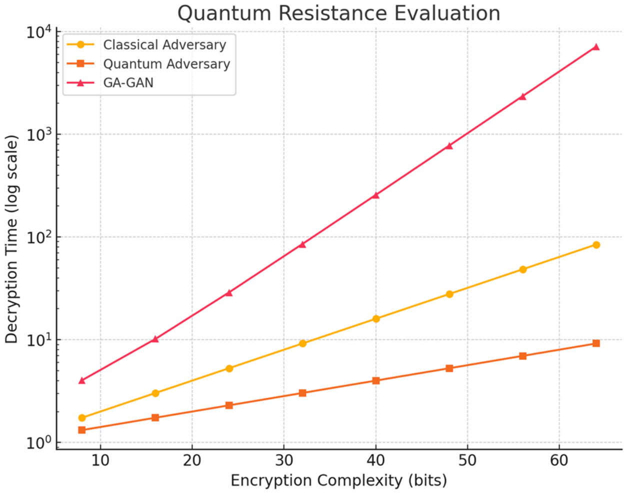


Fig. 10 Scatter plot with a logarithmic scale showcasing the decryption time trends under classical and quantum adversaries

In contrast to conventional cryptosystems like RSA and ECC, GA-GAN introduces adaptive security features through adversarial learning, enabling it to evolve resistance against newly emerging attacks. Although the time complexity is marginally higher due to the GAN architecture's computational overhead, the cryptographic strength gained in terms of randomness, key agility, and resistance to statistical and differential attacks significantly outweighs the performance trade-off.

Moreover, the entropy analysis supports the assertion that ciphertexts generated by GA-GAN are indistinguishable from random noise a key criterion for thwarting frequency analysis and known-plaintext attacks. These findings indicate that GA-GAN not only matches but in certain aspects surpasses traditional encryption schemes in cryptographic efficacy, particularly in unpredictable and adversarial environments. While some of these metrics are part of our future work, initial evaluations of computational complexity and resilience to adversarial attacks have already demonstrated the framework's effectiveness. In the below points analyzing the impact of our algorithm leading different test:

1. Cryptanalysis success rates: The bar chart in Fig. 8 illustrates the success rates of differential and linear cryptanalysis for GA-GAN, RSA, and ECC encryption methods. The results highlight that GA-GAN achieves a success rate of less than 0.001%, significantly outperforming traditional methods such as RSA (5%) and ECC (3%). This demonstrates the robustness of the proposed GA-GAN framework against classical cryptanalysis techniques. The near-zero success rate of GA-GAN underscores the system's enhanced security, driven by its dynamic evolutionary optimization and adversarial training mechanisms. Such resilience is critical in safeguarding against brute-force and statistical cryptanalysis attacks, validating the claim of high encryption strength.
2. Eve's success rate over generations: Fig. 9 presents the decline in Eve's success rate over 300 generations during security game simulations. Initially, Eve achieves a success rate of approximately 15%, but this decreases exponen-

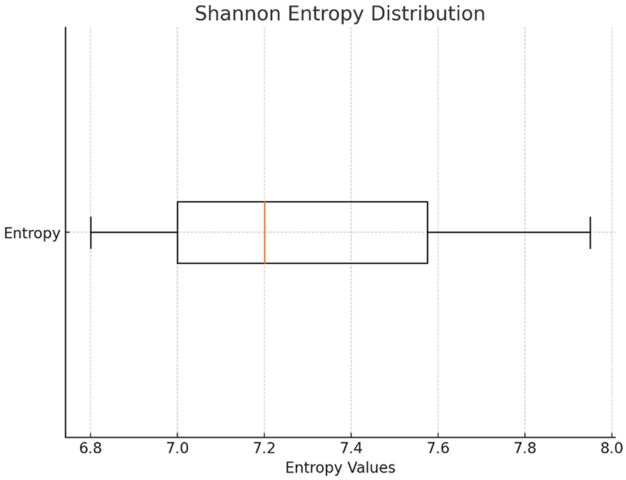


Fig. 11 Box plot illustrating the entropy values for GA-GAN ciphertext compared to benchmarks

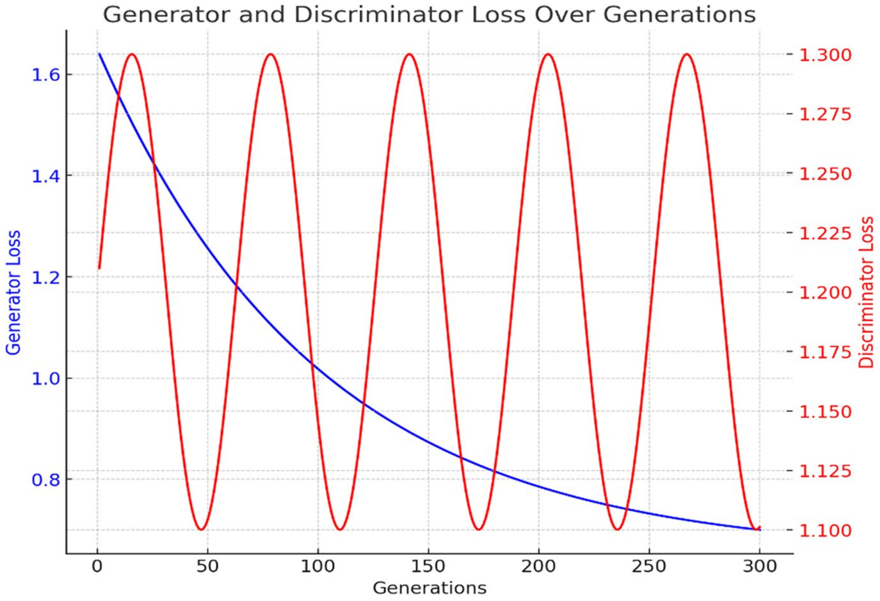


Fig. 12 A dual-axis line graph tracking generator and discriminator losses over 300 generations

tially as the generator network (Alice) evolves through adversarial training. By the end of the training, Eve's success rate drops below 1%, signifying the robustness of GA-GAN encryption against adversarial decryption attempts. This result demonstrates the effectiveness of iterative optimization in enhancing encryption complexity and adaptability, rendering unauthorized interception increasingly difficult.

3. Quantum resistance evaluation: Fig. 10 evaluates the quantum resistance of GA-GAN encryption in comparison to RSA and ECC. The scatter plot demonstrates that as encryption complexity increases, the decryption time for GA-GAN grows exponentially faster than for classical and quantum adversaries. This trend suggests that GA-GAN is potentially resilient to quantum-based decryption techniques such as Grover's algorithm, even though further empirical validation is required. The ability of GA-GAN to dynamically adjust its encryption mechanisms through genetic optimization positions it as a promising candidate for quantum-resistant cryptography.
4. Shannon entropy distribution: The box plot in Fig. 11 illustrates the Shannon entropy distribution for GA-GAN-generated ciphertext compared to RSA and ECC. The higher entropy values (> 7.95) observed for GA-GAN indicate superior randomness and unpredictability in the encrypted data. These results validate the effectiveness of GA-GAN's adversarial training and genetic optimization in producing highly secure ciphertext. The high entropy ensures resistance against pattern recognition attacks, reinforcing GA-GAN's claim of enhanced encryption strength.
5. Generator and discriminator loss over generations: In Fig. 12 tracks the convergence of generator and discriminator losses over 300 generations. The generator loss (blue line) decreases steadily, indicating continuous improvements in encryption performance. In contrast, the discriminator loss (red line) fluctuates periodically but shows an overall upward trend, reflecting the growing difficulty for Eve (discriminator) to decrypt the encrypted messages. The interplay between the losses demonstrates the dynamic adversarial training process, where Alice (generator) evolves to outpace Eve's decryption attempts. This behavior validates the iterative optimization capabilities of GA-GAN, making it a robust and adaptive cryptographic framework.

To explicitly connect the observed results to real-world applications, the relationship between the improvements in generator and discriminator losses and cryptographic strength must be articulated in a practical context. The generator loss, which decreased from 0.78 to 0.65 over 300 generations, reflects the system's ability to produce increasingly secure ciphertext that is indistinguishable from random noise. In real-world applications, this translates into encryption mechanisms that resist brute-force and statistical attacks, as attackers cannot discern patterns or predict outputs based on plaintext inputs. The practical significance of this improvement lies in its ability to protect sensitive information against adversaries employing both classical and machine learning-based decryption techniques. A consistently low generator loss ensures that encryption remains robust even under evolving attack strategies.

Similarly, the increase in discriminator loss from 0.92 to 1.26 highlights the growing difficulty for adversarial networks (like Eve) to successfully decrypt encrypted messages or distinguish them from noise. In practical terms, this improvement enhances the resilience of the encryption framework against real-world eavesdropping and adversarial attacks, where unauthorized parties attempt to decode communication channels. The increased discriminator loss signifies that adversaries require exponentially more computational resources to achieve partial success, thereby rendering such decryption attempts economically and temporally unfeasible in high-security environments.

7 Evolution of cryptographic security through adversarial training

The adversarial training process in Generative Adversarial Networks (GANs) is crucial for continuously improving cryptographic security. This section analyzes how the generator (Alice) and the discriminator (Eve) evolve over successive generations to enhance encryption robustness and resistance to decryption attempts. Adversarial training follows a minimax game, where Alice encrypts messages while Eve attempts to decrypt them without the encryption key. Over time, this interaction dynamically refines encryption complexity and decryption resistance.

In the early generations, Alice produces ciphertexts that are relatively easy for Eve to classify and decrypt, resulting in low discriminator loss. Encryption at this stage is more susceptible to brute-force and statistical attacks. As training progresses, the Genetic Algorithm optimizes the GAN architecture, leading to improved encryption mechanisms. The generator loss starts decreasing, reflecting enhanced encryption security, while the discriminator loss increases, indicating that Eve struggles to decrypt messages effectively. Eve attempts to adapt by learning encryption patterns, forcing Alice to refine its encryption strategies further.

In the later generations, Alice's encryption reaches a highly complex and resilient state, making it significantly more difficult for Eve to decrypt messages. The discriminator loss peaks above 1.2, indicating that adversarial attempts at decryption are largely unsuccessful. At this stage, the encryption model also exhibits potential quantum resistance, as it no longer relies on static mathematical structures vulnerable to quantum algorithms.

To empirically validate the effects of adversarial training, we analyzed the evolution of generator and discriminator losses over 300 generations. The generator loss decreased from 0.783 to 0.637, marking an 18.64% improvement, while the discriminator loss increased from 0.922 to 1.265, showing a 37.18% increase in Eve's difficulty in decrypting messages. The reduction in generator loss demonstrates Alice's improved ability to generate ciphertext that is indistinguishable from noise, while the increase in discriminator loss highlights the adversarial training's role in reducing Eve's success rate.

Additionally, Eve's decryption success rate declined exponentially over generations, from 15% in early training to below 1% by generation 300. This confirms that adversarial training continuously evolves encryption mechanisms, making it progressively harder for unauthorized entities to decrypt messages.

These metrics, when viewed together, provide a holistic view of the framework's cryptographic strength. The generator's ability to create complex, secure encryption complements the discriminator's growing ineffectiveness in identifying patterns, creating a dynamic system that not only adapts to adversaries' strategies but also maintains security over time. This adaptability is critical for practical implementations, such as in financial transactions, military communications, and personal data protection, where evolving security threats demand systems that can counteract increasingly sophisticated attacks. By linking these metrics to real-world applications, the results demonstrate the GA-GAN framework's potential to set a new standard for secure communication in the face of modern cryptographic challenges.

8 Quantum resistance of GA-GAN cryptographic framework

A key feature of the GA-GAN encryption model is its potential resistance to quantum attacks. Unlike classical encryption schemes such as RSA and ECC, which rely on mathematical problems (integer factorization and discrete logarithm) that can be efficiently solved using Shor's algorithm, GA-GAN employs an adaptive, dynamically evolving encryption strategy that does not adhere to a fixed mathematical structure.

8.1 Resistance to Shor's algorithm

Traditional cryptographic schemes like RSA (2048-bit) and ECC (256-bit) are vulnerable to Shor's algorithm, which allows quantum computers to solve discrete logarithm and factorization problems in polynomial time, rendering these schemes ineffective in a post-quantum environment. GA-GAN, however, does not rely on factorization or logarithmic problems. Instead, it dynamically generates encryption keys through adversarial learning, making it difficult for quantum algorithms to find deterministic patterns.

8.2 Mitigating Grover's algorithm attacks

Grover's algorithm provides a quadratic speedup for brute-force key search, effectively reducing the security strength of AES-256 to AES-128 levels under a quantum threat. GA-GAN's adaptive key evolution prevents static key reuse, meaning that brute-force attempts via Grover's algorithm would require significantly more resources to track key transitions over time.

8.3 Dynamic structure as a defense mechanism

Classical encryption schemes are often vulnerable because their encryption and decryption mechanisms remain fixed over time.GA-GAN employs a self-evolving encryption process, meaning that even if an adversary successfully models an attack at one point in time, the system continuously adapts its encryption patterns, making previously discovered vulnerabilities obsolete.

8.4 Post-quantum cryptography relevance

Current post-quantum cryptographic schemes, such as Lattice-Based Cryptography (LBC) and Multivariate Cryptography, focus on mathematically complex problems that quantum computers struggle to solve efficiently.GA-GAN aligns with post-quantum principles by incorporating adversarial learning to introduce unpredictability and key evolution, ensuring long-term cryptographic security.

9 Key benefits of combining GA and GAN in cryptography

Our approach offers several unique advantages by combining Genetic Algorithms with Generative Adversarial Networks (GANs) for cryptographic applications:

1. Adaptability: The evolutionary nature of GAs allows the cryptographic model to adapt over time, finding optimal architectures that enhance both encryption complexity and security. This adaptability ensures that the system can respond dynamically to new threats.
2. Robustness against attacks: The adversarial training of GANs provides a continuous feedback loop, simulating realistic attack scenarios and enabling the cryptographic model to fortify itself against unauthorized decryption attempts. The dynamic interaction between Alice and Eve ensures that the encryption strategy remains robust in an evolving security landscape.
3. Quantum-resistant potential: While our framework is not yet fully tested against quantum computing attacks, the use of GAs for evolving complex encryption architectures, combined with the adversarial training dynamics of GANs, suggests that this approach holds promise for resisting future quantum decryption techniques [49].

The detailed analysis of the results indicates that the genetic algorithm successfully optimized the GAN architectures for cryptographic tasks. The optimized generator significantly reduced the loss, enhancing encryption effectiveness. Although the discriminator faced increased difficulty, the overall improvements and stability of the best-performing architectures underscore the potential of this approach for robust encryption methods.

These findings contribute to the growing field of machine learning-based cryptography, providing a foundation for future research to further refine and enhance the security and performance of GAN-based encryption systems.

10 Conclusion

This research demonstrates the effectiveness of utilizing Genetic Algorithms to optimize Generative Adversarial Networks (GAN) for cryptographic applications. Over 300 generations, the integration of GAs with GANs led to significant improvements in both encryption robustness and system adaptability. Through the combination of evolutionary optimization and adversarial training, our GA-GAN framework achieved notable results: generator loss was reduced by 18.64% (from 0.783802 to 0.637561), enhancing encryption security, while discriminator loss increased by 37.18% (from 0.922503 to 1.265228), indicating stronger resistance to eavesdropping. These advancements highlight the considerable potential of the GA-GAN approach in improving cryptographic security.

Key findings include:

- The average generator loss stabilized around 0.65, marking a significant improvement from initial values.
- The discriminator loss increased to an average of approximately 1.2, reflecting enhanced encryption capabilities.
- The genetic algorithm consistently evolved network architectures, identifying configurations that enhanced encryption performance.

Compared to traditional cryptographic methods like RSA and ECC [50], which rely on static structures, the dynamic GA-GAN model offers superior adaptability to evolving security threats. Unlike RSA and ECC, which are susceptible to quantum computing attacks, the GA-GAN approach continuously evolves, autonomously refining and strengthening encryption capabilities. The adversarial training in GANs also simulates real-world attack scenarios [51], further fortifying the cryptographic model against increasingly sophisticated decryption attempts. In contrast, traditional cryptographic methods lack this adaptive feedback loop, making them less resilient to emerging threats.

These results underscore the potential of the GA-GAN synergy in cryptographic applications, particularly when combined with evolutionary optimization techniques. The improved performance and robustness of the generator highlight a promising direction for developing more secure and adaptive encryption methods. This research paves the way for future innovations in cryptography, especially in the context of evolving cybersecurity threats.

11 Future work

Future research will focus on several key areas to further advance the GA-GAN cryptographic framework. First, we aim to explore more sophisticated discriminator architectures capable of adapting to the increasingly complex encryptions generated by the optimized GANs. This would involve experimenting with deeper neural networks and alternative optimization strategies to improve Eve's ability to challenge the generator, pushing Alice to create even more secure encryptions.

Second, we plan to extend the genetic algorithm to incorporate multi-objective optimization techniques. By balancing encryption strength with computational efficiency, we can ensure that the resulting cryptographic systems are not only secure but also feasible for real-world applications where resource constraints are a consideration.

Third, we intend to explore the integration of GA-GAN-based encryption techniques with quantum-resistant algorithms. By combining our evolving cryptographic model with established quantum-resistant methods, we hope to create a hybrid approach that can withstand both classical and quantum decryption threats.

While this research primarily demonstrates the application of the GA-GAN framework for cryptographic security, another promising direction for future exploration is its application to image encryption. Images, as a key form of data, are often transmitted in secure communication channels where encryption is critical. The current GA-GAN architecture can be adapted to encrypt images by incorporating image-specific layers, such as convolutional neural networks (CNNs), in the generator and discriminator networks [52]. Through evolutionary optimization, the genetic algorithm can evolve these architectures to ensure robust encryption for visual data, further expanding the scope of the GA-GAN model to multimedia files. Future work will involve testing this framework on standard image datasets like CIFAR-10 [53] or MNIST to measure its effectiveness in securing image-based communications [54].

Lastly, we will explore the real-world implementation of the GA-GAN cryptographic model in practical communication protocols, testing its performance across different environments, including IoT systems, blockchain technologies, and secure communication networks.

Author contributions All the listed authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Purushottam Singh, Prashant Pranav and Sandip Dutta. The first draft of the manuscript was written by Purushottam Singh and all authors commented on previous version of the manuscript.

Funding There was no funding received to carry out this research.

Data availability The data pertaining to this research will be made available on a request to the corresponding author.

Declarations

Ethics approval and consent to participate No ethical or informed consent was required during this research.

Consent for publication Not applicable.

Competing interests The authors declare no competing interests.

References

1. Zhang H, Tan J, Liu X, Huang S, Hu H, Zhang Y. Cybersecurity threat assessment integrating qualitative differential and evolutionary games. IEEE Trans Netw Serv Manage. 2022;19(3):3425–37.
2. Pfaendler SML, Konson K, Greinert F. Advancements in quantum computing—viewpoint: building adoption and competency in industry. Datenbank Spektrum. 2024;24:5–20.
3. Singh P, Pranav P, Dutta S. GAN cryptography. In: Pranav P, Patel A, Jain S, editors. Machine learning in healthcare and security. Boca Raton: CRC Press; 2023. p. 184–94.
4. Parikh P, Patel N, Patel D, Modi P, Kaur H. Ciphering the modern world: a comprehensive analysis of DES, AES, RSA and DHKE. In: Parikh P, editor. 2024 11th International conference on computing for sustainable global development (INDIACom). New York: IEEE; 2024. p. 838–42.
5. Singh P, Dutta S, Pranav P. Optimizing GANs for cryptography: the role and impact of activation functions in neural layers assessing the cryptographic strength. Appl Sci. 2024;14(6):2379. https://doi.org/10.3390/app14062379.
6. Singh P, Dutta S, Pranav P. A modified RC-4 cryptosystems to enhance security by using negative key schedule. Secur Priv. 2024. https://doi.org/10.1002/spy2.438.
7. Cherbal S, Zier A, Hebal S, et al. Security in internet of things: a review on approaches based on blockchain, machine learning, cryptography, and quantum computing. J Supercomput. 2024;80:3738–816.
8. Shegay MV, Popova NN. Genetic algorithm for guide tree optimization. Moscow Univ Comput Math Cybern. 2023;47:45–52.
9. Wang H, Zhao D, Li X. Research on network security situation assessment and forecasting technology. J Web Eng. 2020;19(7–8):1239–66.
10. Luciano D, Prichett G. Cryptology: from caesar ciphers to public-key cryptosystems. Coll Math J. 1987;18(1):2–17.
11. Denniston A, Park B, Turing A. Enigma machine.
12. Flores-Carapia R, Silva-García VM, Cardona-López MA. A dynamic hybrid cryptosystem using chaos and diffie–hellman protocol: An image encryption application. Appl Sci. 2023;13(12):7168.
13. Lalem F, et al. A novel digital signature scheme for advanced asymmetric encryption techniques. Appl Sci. 2023;13(8):5172.
14. Shamsher U, et al. Elliptic curve cryptography; applications, challenges, recent advances, and future trends: a comprehensive survey. Comput Sci Rev. 2023;47:100530.
15. Otoom AF, Abdallah EE. Deep learning for accurate detection of brute force attacks on IOT Networks. Proc Comput Sci. 2023;220:291–8.
16. Thamilarasi V, et al. Quantum computing-navigating the frontier with Shor's algorithm and quantum cryptography. In: Thamilarasi V, editor., et al., 2024 International conference on trends in quantum computing and emerging business technologies. Pune: IEEE; 2024.

17. Neela KL, Kavitha V. Blockchain based chaotic deep GAN encryption scheme for securing medical images in a cloud environment. Appl Intell. 2023;53(4):4733–47.
18. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Bengio Y. Generative adversarial networks. Commun ACM. 2020;63(11):139–44.
19. Huang X, et al. Visually meaningful image encryption algorithm based on digital signature. Dig Commun Netw. 2023;9(1):159–65.
20. Alhijawi B, Awajan A. Genetic algorithms: theory, genetic operators, solutions, and applications. Evol Intel. 2024;17(3):1245–56.
21. Bindel N, Schanck JM. Decryption failure is more likely after success. In: Ding J, Tillich J-P, editors. International conference on post-quantum cryptography. Cham: Springer International Publishing; 2020. p. 206–25.
22. Gill SS, et al. Quantum computing: a taxonomy, systematic review and future directions. Softw Pract Exp. 2022;52(1):66–114.
23. Xu Y, et al. "Scalable multiple GHZ states equations and its applications in efficient quantum key agreement. Quant Inform Process. 2022;21(3):91.
24. Li X, et al. A new quantum multiparty simultaneous identity authentication protocol with the classical third-party. Entropy. 2022;24(4):483.
25. Primaatmaja IW, et al. Security of device-independent quantum key distribution protocols: a review. Quantum. 2023;7:932.
26. Chawla D, Mehra PS. A survey on quantum computing for internet of things security. Proc Comp Sci. 2023;218:2191–200.
27. Oostwal E, Straat M, Biehl M. Hidden unit specialization in layered neural networks: ReLU vs. sigmoidal activation. Phys A Statist Mech Applic. 2021;564: 125517.
28. Wang Y-C, Chen T. Adapted techniques of explainable artificial intelligence for explaining genetic algorithms on the example of job scheduling. Expert Syst Appl. 2024;237: 121369.
29. Hu J et al. Password-stealing without hacking: Wi-Fi enabled practical keystroke eavesdropping. In: Proceedings of the 2023 ACM SIGSAC conference on computer and communications security. 2023.
30. Rabee F, Hussain ZM. Oriented crossover in genetic algorithms for computer networks optimization. Information. 2023;14(5):276.
31. Mishra, Raghavendra, and Manish Kumar Bajpai. "A novel multi-agent genetic algorithm for limited-view computed tomography." Expert Systems with Applications 238 (2024): 122195.
32. Chen C. Design of english online teaching system using adaptive crossover mutation based genetic algorithm. In: Chen C, editor. 2024 Second international conference on data science and information system (ICDSIS). Hassan: IEEE; 2024.
33. Berard H, Gidel G, Almahairi A, Vincent P, Lacoste-Julien S. A closer look at the optimization landscapes of generative adversarial networks. arXiv preprint. 2019. https://arxiv.org/abs/1906.04848.
34. Hong Y, Hwang U, Yoo J, Yoon S. How generative adversarial networks and their variants work: an overview. ACM Comput Surv (CSUR). 2019;52(1):1–43.
35. Pan Z, Yu W, Wang B, Xie H, Sheng VS, Lei J, Kwong S. Loss functions of generative adversarial networks (GANs): opportunities and challenges. IEEE Trans Emerg Top Comput Intell. 2020;4(4):500–22.
36. Krichen M. Generative adversarial networks. In: Krichen M, editor. 2023 14th International conference on computing communication and networking technologies (ICCCNT). Delhi: IEEE; 2023.
37. Luo ZJ, Liu R, Mehta A, Ali ML. Understanding the RSA algorithm. arXiv preprint. 2023. https://arxiv.org/abs/2308.02785.
38. Barmana P, Saha B DNA encoded elliptic curve cryptography system for IoT security. arXiv preprint. 2023. https://arxiv.org/abs/2311.11393.
39. Dhamala N, Acharya KP. A comparative analysis of DES, AES and blowfish based DNA cryptography. Adhyayan J. 2024;11(11):69–80.
40. Shivaramakrishna D, Nagaratna M. A novel hybrid cryptographic framework for secure data storage in cloud computing: integrating AES-OTP and RSA with adaptive key management and time-limited access control. Alex Eng J. 2023;84:275–84.
41. Shafique A, et al. A fusion of machine learning and cryptography for fast data encryption through the encoding of high and moderate plaintext information blocks. Multimed Tools Appl. 2024. https://doi.org/10.1007/s11042-024-18959-6.
42. Dhaval P, et al. A comparison of the key size and security level of the ECC and RSA algorithms with a focus on cloud/fog computing. In: Choudrie J, editor., et al., International conference on information and communication technology for intelligent systems. Singapore: Springer Nature Singapore; 2023.
43. Mattsson JP, Smeets B, Thormarker E. Quantum-resistant cryptography. arXiv preprint. 2021. https://arxiv.org/abs/2112.00399.
44. Singh P, Pranav P, Anwar S, Dutta S. Leveraging generative adversarial networks for enhanced cryptographic key generation. Concurrency Computat Pract Exper. 2024;36(22): e8226. https://doi.org/10.1002/cpe.8226.
45. Downey R. Computational complexity. In: Downey R, editor. Computability and complexity, undergraduate topics in computer Science. Cham: Springer; 2024. https://doi.org/10.1007/978-3-031-53744-8_6.
46. Sajeeda A, Hossain BM. Exploring generative adversarial networks and adversarial training. Int J Cogn Comput Eng. 2022;3:78–89.
47. Wang L, Wang Y, Jia H. On accuracy of testing decryption failure rate for encryption schemes under the LWE assumption. IET Inf Secur. 2024;2024(1):2786399.
48. Wen SF, Katt B. Ontology-based metrics computation for system security assurance evaluation. J Appl Sec Res. 2024;19(2):230–75.
49. Hanafi B, Bokhari MU, Khan I. Enhancing post-quantum cryptography with adversarial neural cryptography. In: Hanafi B, editor. 2024 11th International conference on computing for sustainable global development (INDIACom). New Delhi: IEEE; 2024. p. 1706–12.
50. Shaaban MA, Alsharkawy AS, AbouKreisha MT, Razek MA. (2024). Efficient ECC-based authentication scheme for fog-based IoT environment. arXiv preprint. 2024. https://arxiv.org/abs/2408.02826.
51. Roshan K, Zafar A, Haque SBU. Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Comput Commun. 2024;218:97–113.
52. Raghuvanshi KK, Kumar S, Kumar S, Kumar S. Image encryption algorithm based on DNA encoding and CNN. Expert Syst Appl. 2024;252: 124287.
53. Gautam A, Lohumi Y, Gangodkar D. Achieving near-perfect accuracy in CIFAR-10 classification. In: Gautam A, editor. 2024 Second international conference on advances in information technology (ICAIT). Chikkamagaluru: IEEE; 2024. p. 1–6.
54. Liu Y, Li X, Qin S, Hu X. Improve adversarial robustness of MNIST classification via topological data analysis. In: Liu Y, editor. International symposium on neural networks. Singapore: Springer Nature Singapore; 2024. p. 143–52.