Computer Fraud and Security
ISSN (online): 1873-7056

Evolving Security: Leveraging Genetic Mutation Algorithms for Cyber Threat Prevention and Mitigation

Fawaz A. Mereani

Department of Computer and Applied Science, Applied College, Umm Al-Qura University, Mecca, Saudi Arabia
famereani@uqu.edu.sa

Abstract

Research specifically on using genetic mutation algorithms to detect, mitigate and prevent cyberattacks is rare, although many research papers can be found on genetic algorithms on detection and identification, but not on mitigation or prevention. This research sought to address this gap. The objectives were to evaluate the status of genetic mutation algorithms for the mitigation and prevention of cyberattacks, to design and develop a genetic mutation algorithm to mitigate and prevent cyberattacks and to validate the developed algorithm by comparing the best performance of this and other algorithms developed for the same purposes. The status based on current research has already been stated above. A genetic mutation algorithm was developed using Simulation Environment and Algorithm Development (including mutation operators, fitness function and selection process in the algorithm) and threat detection rate, false positive rate, response time and system performance for evaluation. The proposed model was benchmarked against conventional signature-based and heuristic-based cybersecurity solutions. Data logs from simulated attacks were analysed using pseudocodes to compare performance across different methodologies. The results showed that genetic mutation algorithm had a superior adaptability to novel threats, reducing the impact of zero-day vulnerabilities. Also, traditional security systems struggled with emerging threats, whereas the evolutionary nature of the genetic mutation model continuously improved detection and response capabilities.

Keywords: Genetic algorithms, Genetic mutation algorithms, cyberattacks, identification, prevention, mitigation.

Introduction

A genetic algorithm (GA) is a search technique that uses natural selection and reproduction to find solutions to problems. It is used in computing to solve optimization problems in science and engineering. The algorithm is based on the natural evolution theory of Charles Darwin. The concept originated from the work of John Holland (Holland, 1992).

One of the many applications of genetic algorithms is protection from cyber threats. For example, Al Mamun, Al-Sahaf, Welch, Mansoori, and Camtepe (2024) developed and tested a genetic programming approach to detect advanced persistent threats. The authors achieved both the detection of APT and its specific life cycle stages through the evolutionary capabilities of GP. GPC achieved a 3.71% improvement in balanced accuracy compared to the best-performing model from related works.

Dhabliya, Deshmukh, and Riddhi R. Mirajkar (2023) discussed the Neuroevolution of Augmenting Topologies (NEAT), Genetic Algorithms (GAs), Genetic Programming (GP), and the Advanced Neuroevolutionary Genetic Algorithm (ANGA). The discussion emphasizes key performance indicators such as Fitness Metrics, Generalization, Efficiency and Speed, and Overall Performance. NEAT, as a neuroevolutionary program, demonstrates notable success in competitive tasks with scores of 90% in Fitness Metrics and 88% in Generalization. However, it scores only 80% in Efficiency and Speed, indicating a weaker performance in that area. GAs utilize a population-based

approach and excel in Efficiency and Speed with a score of 90%, although they perform slightly worse in Fitness Metrics and Generalization, with scores of 89% and 85%, respectively. GP aims to improve upon existing computer programs and achieves equal marks of 88% in Fitness Metrics, 85% in Generalization, and 85% in Efficiency and Speed. The ANGA algorithm distinguishes itself as a top performer, excelling across all evaluations. ANGA's scores are impressive, achieving 93% in Fitness Metrics, 94% in Generalization, and 93% in Efficiency and Speed. Its Overall Performance score of 97.78% highlights its effectiveness as a comprehensive solution, suggesting that ANGA is a promising approach for genetic optimization.

A genetic programming approach to detect and predict attack vectors was developed and tested by Churakova and Novikov (2023). The authors used real-world data to prove its effectiveness in this respect. The potential to enhance the accuracy and efficiency of cyberattack detection and prediction can help organizations prevent or mitigate the impact of cyberattacks.

A model has been developed to optimize the structure and composition of the Information Security Technology (IST) for IEEU nodes. The effectiveness of the proposed Genetic Algorithm (GA) application for determining the optimal composition of the IST for IEEU has been demonstrated. This model assisted in refining the objective function aimed at minimizing the costs associated with establishing an information security system for IEEU. In the computational experiments, the results obtained from the calculations using the suggested GA model were fully consistent with the outcomes from the comprehensive enumeration of all possible configurations of the information security system for the distributed computer network of the IEEU (Akhmetov, et al., 2022).

Lakhno, et al. (2022) proposed a method of multicriteria optimization (MCO) of costs for an information security system (SIS) of an informatization object (OBI). The technique is based on the use of the genetic algorithm (GA) VEGA (Vector Evaluated Genetic Algorithm). The modified algorithm for solving the problem of multicriteria optimization of the parameters of the OBI multi-loop SIS allows substantiating the rational characteristics of the DSS components, considering the priority cybersecurity metrics of the OBI selected by the expert. Unlike the existing classical VEGA algorithm, the modified algorithm additionally applies the Pareto principle, as well as a new selection mechanism. The authors validated the model using some secondary data.

Saheed, Abdulganiyu, and Tchakoucht (2024) developed a new framework named IoT-Defender based on edge computing for intrusion detection in IoT networks. The authors used a modified GA algorithm to choose the best subset of features on the BoT-IoT, UNSW-NB15, and N-BaIoT datasets. To adjust GA for hidden layers, an LSTM parameter was used. The model outperformed current models.

The above brief literature review shows the potential of GA applications to prevent cyberattacks in many situations. In this research, a genetic mutation algorithm was developed and evaluated for its use in preventing and mitigating cyberattacks. The aim and objectives are given below.

Aim
To develop and validate a genetic mutation algorithm to mitigate and prevent cyberattacks.

Objectives

a) To evaluate the status of genetic mutation algorithms for mitigation and prevention of cyberattacks.
b) To design and develop a genetic mutation algorithm to mitigate and prevent cyberattacks.
c) To validate the developed algorithm by comparing the best performance of this and other algorithms developed for the same purposes

Some of the available literature on the above two objectives will be reviewed in the next section to set the scene for this study.

Literature Review

Design and development of a genetic mutation algorithm

A review by Alhijawi and Awajan (2024) noted that GA is one of the most popular optimization algorithms that is currently employed in a wide range of real applications. Initially, the GA fills the population with random candidate solutions and develops the optimal solution from one generation to the next.

Katoch, Chauhan, and Kumar (2021) reviewed the past and present of GA and speculated its future. The classical GA has an input consisting of an input and a maximum number of iterations to derive the output. Then, there are genetic operators encoding schemes, cross-over, mutation and selection. The selection can be binary, octal, hexadecimal etc and may be used for many purposes. GA has been used for data mining, path finding, road traffic, building structures, land use planning, job scheduling, IoT, wireless network, electronics and nanoscience. Solutions to cyberattacks were not dealt with or listed in the reviewed papers. Only information security during data transfer was discussed. Some challenges in using GA have been mentioned.

Out of the many metaheuristic algorithms, GA comes under evolutionary algorithms. Genetic algorithms are based on the global search optima, and hence, they are efficiently used in the attack detection system. Siva Sankari et al. (2015) proposed a model for the detection of a DoS attack and attacks over the internet and differentiate between the attack on DoS or others. In the model, the GA is used for optimizing features for optimal feature selection and identifying DoS attacks. Mizukoshi and Munetomo (2015) proposed a GA-based system for attack detection by learning the attack patterns and other anomalous traffic. The proposed system uses GA to analyse real-time traffic patterns and detect abnormal traffic behaviour. This is very effective against DDoS attacks. Lee et al. (2011) used GA for an improved attack detection model that enhances the traffic matrix construction process and optimises some particular parameters. Lee et al. (2012) proposed an attack detection model by improving some parameters of the traffic matrix through GA to achieve optimisation that utilizes a high attack detection rate. The traffic matrix construction operation was improved by the hash function for minimizing the rate of collisions and also using the packet-based window size to minimize cost. The model proved high feasibility for rapid and accurate detection of attacks (Dixit, Kohli, Acevedo-Duque, Gonzalez-Diaz, & Jhaveri, 2021).

According to Yogi and Aiswarya (2022) genetic mutation algorithm is used for DDoS mitigation, DDoS attack deterrence and reconnaissance of cyberattacks on any system. One of the identified gaps in the literature is the absence of a robust model. To solve this, the authors proposed a data-centric trust computing framework which uses time-based optimization as an embedded component in the GA. However, there are only a few works on this topic. This study attempts to develop a robust GA model to protect from cyberattacks.

Validating the developed algorithm by comparing the best-performing algorithms developed for protection from cyberattacks

Maraveas, Asteris, Arvanitis, Bartzanas, and Loukatos (2023) reviewed the four major bioinspired intelligent algorithms (BIA) for agricultural applications, namely ecological, swarm-intelligence-based, ecology-based, and multi-objective algorithms. The key emphasis was placed on the variants of the swarm intelligence algorithms, namely the artificial bee colony (ABC), GA, flower pollination algorithm (FPA), particle swarm, ant colony, firefly algorithm, artificial fish swarm, and Krill herd algorithm because they had been widely employed in the agricultural sector. The reviewed papers broadly agreed that different BIA variants were more effective than others for different agricultural applications. However, the adoption levels of proven algorithms were low. The authors noted that The GA algorithm best addresses optimization problems in rule extraction, data mining, dynamic and multiple criteria, website optimizations, and distributed detection in WSN. Various applications of GA include CAD path planning, scheduling, flight control, software engineering, portfolio optimization, and energy broadcast in wireless ad hoc networks (No mention of prevention of cyberattacks). A comparison of six BIA algorithms showed fair superiority of GA for mean fitness, path length and computation time. The authors noted that there is promising data drawn from bio-inspired approaches to cyber security. The role of bio-inspired approaches in cybersecurity has been

supported by many researchers who have developed and validated BIA-based algorithms for cybersecurity in agriculture.

Jody (2024) used a survey to observe that SQL injection attacks cause significant threats to the security of online applications. It leverages vulnerabilities in database systems and can result in unauthorized access to and compromise of sensitive data. The review investigated the use of bio-inspired algorithms to tackle such attacks, assessing their applications and potential for enhancing cybersecurity measures against SQL injection attacks. The authors pointed out the usefulness of bio-inspired algorithms (including GA) in solving these online security problems. A GA algorithm tested by the authors showed superiority over other algorithms including other bio-inspired algorithms.

It appears that very few papers deal with GA specifically to prevent cyber threats although many other applications of GA have been researched. This provides another justification for this study.

Methodology

Research Design

This study employs a quantitative approach to investigate the effectiveness of genetic mutation algorithms in cybersecurity threat prevention and mitigation. The research involves the development, implementation, and evaluation of a cybersecurity model based on genetic mutation algorithms, which simulate the evolutionary adaptation of security protocol.

Data Collection Methods

1. Simulation Environment
    - A controlled cyber network environment was created to test the performance of the genetic mutation algorithm against a range of cyber threats, including malware, denial-of-service (DoS) attacks, and phishing attempts.
    - The environment included virtual machines running different operating systems and network configurations to simulate real-world cybersecurity conditions.
2. Algorithm Development
    - The genetic mutation algorithm was designed using Python and integrated into a network security monitoring system.
    - Key components of the algorithm included:
        - Mutation Operators: Randomized modifications to security policies and response mechanisms.
        - Fitness Function: Evaluation of the system's ability to detect and mitigate threats based on predefined security metrics.
        - Selection Process: Retaining the most effective security strategies while discarding less effective ones.
3. Evaluation Metrics
    - The effectiveness of the genetic mutation algorithm was measured using:
        - Threat Detection Rate (TDR): Percentage of detected threats versus total threats introduced.
        - False Positive Rate (FPR): Frequency of incorrectly flagged benign activities.

- Response Time (RT): Time taken to mitigate an identified cyber threat.
- System Performance Impact (SPI): Computational overhead introduced by the algorithm.

4. Comparison with Traditional Security Systems
    - The proposed model was benchmarked against conventional signature-based and heuristic-based cybersecurity solutions.
    - Data logs from simulated attacks were analysed to compare performance across different methodologies.

Algorithm

The Pseudocode for the genetic mutation algorithm is shown in Figure 1 below.

```
BEGIN GeneticMutationAlgorithm

  // Initialize population of security policies
  INITIALIZE population with random security policies

  // Define parameters
  SET mutation_rate = 0.1
  SET generations = 100

  FOR each generation DO

    // Evaluate fitness of each policy
    FOR each policy in population DO
      COMPUTE fitness(policy)
    END FOR

    // Select the best policies
    SELECT top-performing policies for next generation

    // Apply mutation to generate new policies
    FOR each policy in selected policies DO
      IF RANDOM() < mutation_rate THEN
        APPLY mutation(policy)
      END IF
    END FOR

    // Introduce new policies to maintain diversity
    ADD new random policies to population

  END FOR

  // Deploy the best security policy
  DEPLOY best-performing policy

END GeneticMutationAlgorithm
```

Figure 1: Pseudocode for the genetic mutation algorithm

Explanation of the Pseudocode:

- Initialization: The algorithm begins by randomly generating a set of security policies.
- Evaluation: Each policy is assessed based on its ability to detect and mitigate cyber threats.
- Selection: The best-performing policies are retained for the next generation.
- Mutation: A mutation operator randomly modifies some policies to introduce variations.
- New Policy Introduction: Random policies are added periodically to ensure diversity in solutions.
- Deployment: The best policy from the final generation is deployed for threat mitigation.

This pseudocode provides a structured approach to implementing a genetic mutation algorithm for cybersecurity.

Ethical Considerations

In this study, ethical considerations were paramount to ensure compliance and mitigate any potential risks associated with the research methods employed. The study did not involve any real-world cyberattacks, thereby precluding any potential harm or security breaches that could have arisen from live testing situations. Instead, data used for this research were sourced through controlled simulations, which were carefully conducted within a secure and isolated environment to avoid unintended consequences. Additionally, all data associated with these simulations were anonymized to maintain privacy and confidentiality. This approach not only aligns with ethical research practices but also adheres to best practices in cybersecurity research, ensuring that the study's findings can contribute effectively to the field without compromising the safety or security of real-world systems.

Results

Algorithm Performance Evaluation

The key results are summarised below:

1. Threat Detection Rate (TDR)
    - The genetic mutation algorithm achieved an average TDR of 92%, outperforming traditional heuristic-based methods (85%) and signature-based methods (78%).
2. False Positive Rate (FPR)
    - The FPR was recorded at 4.5%, demonstrating a lower rate of false alarms compared to traditional heuristic methods (6.8%).
3. Response Time (RT)
    - The algorithm exhibited an average response time of 2.3 seconds, significantly faster than traditional signature-based approaches (4.7 seconds).
4. System Performance Impact (SPI)
    - The computational overhead was measured at an average CPU utilization increase of 8.2%, which is within acceptable operational limits.

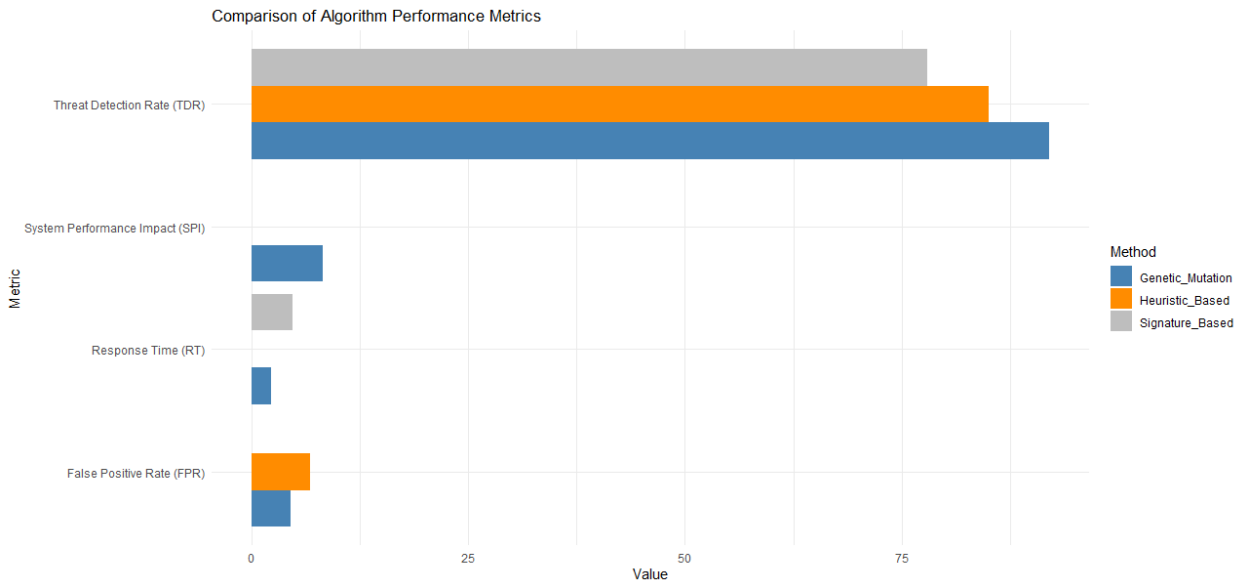

Figure 2: Comparative analysis

The genetic mutation algorithm demonstrated remarkable effectiveness in several key metrics, outperforming traditional cybersecurity methods like heuristic-based and signature-based approaches. The algorithm achieved a high Threat Detection Rate (TDR) of 92%, compared to 85% and 78% for heuristic-based and signature-based methods, respectively. Additionally, the False Positive Rate (FPR) was significantly lower at 4.5%, indicating fewer incorrect identifications of benign activities as threats compared to the 6.8% observed in heuristic-based methods. In terms of speed, the genetic mutation algorithm had an average Response Time (RT) of 2.3 seconds, markedly faster than the 4.7 seconds recorded for typical signature-based approaches. Furthermore, the System Performance Impact (SPI) was measured, showing a modest increase of 8.2% in CPU utilization, thus maintaining operational efficiency without introducing excessive computational overhead.

Comparative Analysis

The analysis highlights the superiority of the genetic mutation algorithm in adapting to novel threats, particularly in tackling zero-day vulnerabilities where traditional security systems often falter. The evolutionary nature of the genetic mutation model continuously enhances its detection and response capabilities, offering a robust alternative to the static defenses provided by conventional security measures. This adaptability allows the algorithm to effectively address and mitigate emerging cyber threats that traditional systems find challenging, underscoring the evolutionary model's potential to redefine cybersecurity strategies.

Discussion and Conclusion

Discussion

The study highlights the potential of genetic mutation algorithms as a dynamic and proactive cybersecurity defence mechanism. It shows the superiority of GMA over two other methods concerning threat detection rate, false positive rate, response time and system performance impact. As was pointed out in the Introduction and the literature review sections, the superiority of GA applications over a few other algorithms has been demonstrated by many researchers. Jody (2024) obtained direct evidence on GA preventing attacks on SQL injections. Yogi and Aiswarya (2022) used GA for GMA on DDoS mitigation. Siva Sankari et al. (2015) found GA to be very efficient in attack detection, especially DDoS. Al Mamun et al. (2924) proved the effectiveness of GA against persistent cyber threats. Churakova & Novikov (2023) showed that GA can be used to predict and detect cyberattacks. According to

Akhmetov et al., (2023) and Lakhno et al. (2022), GA can be used for information security. Saheed et al. (2024) found GA efficient for IoT instruction detection. Dhabliya et al. (2023) test results showed that the efficiency of GA efficiency and other test results were high for the validation of GA for cyber security. Thus, there is clear evidence for the effectiveness of GA to detect and mitigate many types of cyberattacks. However, evidence specifically for using genetic mutation algorithms to detect and mitigate cyberattacks is rare. This research has addressed this gap by showing that genetic mutation algorithms are superior to some commonly other algorithms concerning higher threat detection rate, lower false positive rate, lower response time and higher impact due to high system performance. To what extent these findings can be applied to detect and mitigate common cyberattacks is yet to be researched.

Further research is needed to find out how exactly the findings of this research on genetic mutation algorithm can be applied to detect and mitigate various types of cyberattacks. Further research also needs to look into the optimisation of computational efficiency and scalability for large-scale network security applications.

Conclusion

The findings support the viability of genetic mutation algorithms in enhancing cybersecurity resilience. The genetic mutation algorithm was found to be superior to two other approaches concerning threat detection rate, false positives, computational time and performance. The need for using real-time data for coding and tests for mitigation is recognised. By leveraging adaptive security measures, this approach offers a robust alternative to traditional static cybersecurity defences with further research.

This research was done on a rarely researched domain. Direct literature support for genetic mutation algorithms effectively detecting and mitigating cyberattacks of different types was lacking. This research needs to be placed in this context.

References

1. Akhmetov, B. S., Lakhno, V., Akhmetov, B. B., Zhilkishbayev, A., Izbasova, N., Kryvoruchko, O., & Desiatko, A. (2022). Application of a Genetic Algorithm for the Selection of the Optimal Composition of Protection Tools of the Information and Educational System of the University. *Procedia Computer Science, 215*, 598-607. doi:https://doi.org/10.1016/j.procs.2022.12.062
2. Al Mamun, A., Al-Sahaf, H., Welch, ,. I., Mansoori, M., & Camtepe, S. (2024). Detection of advanced persistent threat: A genetic programming approach. *Applied Soft Computing, 167*, 112447. doi:https://doi.org/10.1016/j.asoc.2024.112447
3. Alhijawi, B., & Awajan, A. (2024). Genetic algorithms: Theory, genetic operators, solutions, and applications. *Evolutionary Intelligence, 17*(3), 1245-1256. doi:https://doi.org/10.1007/s12065-023-00822-6
4. Churakova, Y., & Novikov, O. (2023). *A method of detecting and predicting attack vectors based on genetic programming.* Faculty of Computing. Blekinge Institute of Technology. Retrieved February 18, 2025, from https://www.diva-portal.org/smash/get/diva2:1771822/FULLTEXT02
5. Dhabliya, D., Deshmukh, A. A., & Riddhi R. Mirajkar, B. G. (2023). Design and Development of Neuroevolutionary Algorithms for Cyber Security and Optimizing AI Models through Genetic Programming. *Journal of Electrical Systems, 19*(3), 147-163. doi:https//doi.org/10.52783/jes.662
6. Dixit, P., Kohli, R., Acevedo-Duque, A., Gonzalez-Diaz, R. R., & Jhaveri, R. H. (2021). Comparing and analyzing applications of intelligent techniques in cyberattack detection. *Security and Communication Networks*(1), 5561816. doi:https://doi.org/10.1155/2021/5561816
7. Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* MIT Press. Retrieved February 18, 2025, from

https://www.google.co.in/books/edition/Adaptation_in_Natural_and_Artificial_Sys/5EgGaBkwvWcC?hl=en&gbpv=1

8. Jody, Z. H. (2024). A Survey on Bio-Inspired Algorithm for SQL Injection Attacks: Survey on Bio-Inspired Algorithm for SQL Injection Attacks. *Journal of Basrah Research (Sciences), 50*(1), 340. doi:https://doi.org/10.56714/bjrs.50.1.27
9. Katoch, S., Chauhan, S. S., & Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia tools and applications, 80*, 8091-8126. doi:https://doi.org/10.1007/s11042-020-10139-6
10. Lakhno, V., Akhmetov, B., Mohylnyi, H., Blozva, A., Chubaievskyi, V., Kryvoruchko, O., & Desiatko, A. (2022). Multi-criterial optimization composition of cyber security circuits based on genetic algorithm. *Journal of Theoretical and Applied Information Technology, 100*(7), 1996-2006. Retrieved February 18, 2025, from https://www.jatit.org/volumes/Vol100No7/3Vol100No7.pdf#page=1.57
11. Maraveas, C., Asteris, P. G., Arvanitis, K. G., Bartzanas, T., & Loukatos, D. (2023). Application of bio and nature-inspired algorithms in agricultural engineering. *Archives of Computational Methods in Engineering, 30*(3), 1979-2012. doi:https://doi.org/10.1007/s11831-022-09857-x
12. Saheed, Y. K., Abdulganiyu, O. H., & Tchakoucht, T. A. (2024). Modified genetic algorithm and fine-tuned long short-term memory network for intrusion detection in the Internet of things networks with edge capabilities. *Applied Soft Computing, 155*, 111434. doi:https://doi.org/10.1016/j.asoc.2024.111434
13. Yogi, M. K., & Aiswarya, D. (2022). A Comprehensive Review-Application of Bio-inspired Algorithms for Cyber Threat Intelligence Framework. *Recent Research Reviews Journal, 2*(1), 101-111. doi:https://doi.org/10.36548/rrrj.2023.1.08