International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

Efficient Fault Tolerance Methodology in Fanet Using Aco and Ml Techniques

Pooja sri G[1], Nuha Fathima N[2], Abinaya B[3]
[1,2]UG - Electronics and Communication Engineering, Easwari Engineering College, Bharathi Salai, Ramapuram, Chennai, India.
[3]Assistant Professor, Easwari Engineering College, Bharathi Salai, Ramapuram, Chennai, India.
Emails: *g.poojasri02@gmail.com[1], nuhanf@gmail.com[2], abinaya.b@eec.srmrmp.edu.in[3].*

Abstract
An innovative approach is presented in this study to enhance the performance of Ant Colony Optimization (ACO), a type of Bio-Inspired Algorithm (BIA), by integrating machine learning (ML) techniques for fault prediction. The goal is to address the challenges of high end-to-end delay and susceptibility to faults in traditional ACO implementations by leveraging ML methods. Through the application of ML techniques to optimize ACO efficiency and anticipate faults using the Random Forest model, significant reductions in end-to-end delay and improvements in system survivability are achieved. Additionally, the utilization of Least Absolute Shrinkage and Selection Operator (LASSO) feature selection streamlines the optimization process and enhances overall performance. Experimental results demonstrate the superiority of the proposed ML-enhanced ACO approach, indicating its potential for real-world applications in optimization problems.
Keywords: *ACO - Ant Colony Optimization; BIA - Bio Inspired Algorithms; ML - Machine Learning; SVM - Support Vector Machine; UAVs - Unmanned Aerial Vehicles.*

1. Introduction

In recent years, the field of optimization has witnessed significant advancements, driven by the integration of bio-inspired algorithms (BIAs) with machine learning (ML) techniques. Ant Colony Optimization (ACO) is a well-known method that draws inspiration from the foraging habits of ants. Routing, scheduling, and graph-based activities are just a few of the combinatorial optimization issues that have been effectively solved with ACO. However, traditional ACO implementations face challenges related to high end-to-end delay and vulnerability to faults. These limitations hinder their widespread adoption in real-world scenarios. To address these issues, researchers have explored novel approaches that leverage ML methods to enhance ACO's performance and robustness.

1.1 The Objectives of Our Work are Twofold

Fault Anticipation (Wang W., et al., 2023;) and System Survivability: We seek to anticipate faults in ACO execution. By employing ML-based fault prediction models, we can enhance system survivability and minimize the impact of unexpected failures.

Implementation in ACO logic: When a faulty node is encountered or failure occurs between one node to another node, Implementation of ML model in ACO helps to update ACO logic according to the fault prediction. Our approach combines the Random Forest model for fault prediction and Lasso feature selection to streamline the optimization process [1]. The proposed ML-ACO demonstrates promising results, highlighting its potential for real-world applications in optimization problems.

1.2 Motivation

This research stems from the desire to enhance the performance and applicability of Ant Colony Optimization (ACO) in real-world scenarios. Traditional ACO implementations, while effective in solving combinatorial optimization problems, often suffer from high end-to-end delay and vulnerability to faults, limiting their practical utility. These shortcomings can hinder timely decision-making and compromise system reliability, highlighting the need for innovative

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

approaches to improve ACO's efficiency and robustness. By integrating machine learning (ML) techniques with ACO, we aim to address these challenges and unlock new possibilities for [2] optimization in diverse fields. ML offers the potential to predict and mitigate faults, reduce end-to-end delay, and optimize ACO parameters, ultimately enhancing its performance and expanding its range of applications.

1.3 Background

In a comparative study, based on Fault Tolerance, of BIAs including Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Bee Colony Optimization (BCO), and Firefly Algorithm (FA), ACO emerged as the top-performing algorithm. The evaluation criteria encompassed essential parameters such as throughput, packet dropped ratio, packet delivery ratio, and the number of hops [3]. ACO's superiority in these metrics underscores its efficacy in addressing combinatorial optimization challenges, positioning it as a leading choice for real-world applications. A thorough understanding of each BIAs performance may be obtained by examining the metrics Packet Delivery Ratio (PDR), Packet Dropped Ratio, which is expressed in Figure 1, Throughput Which is expressed in Figure 2.

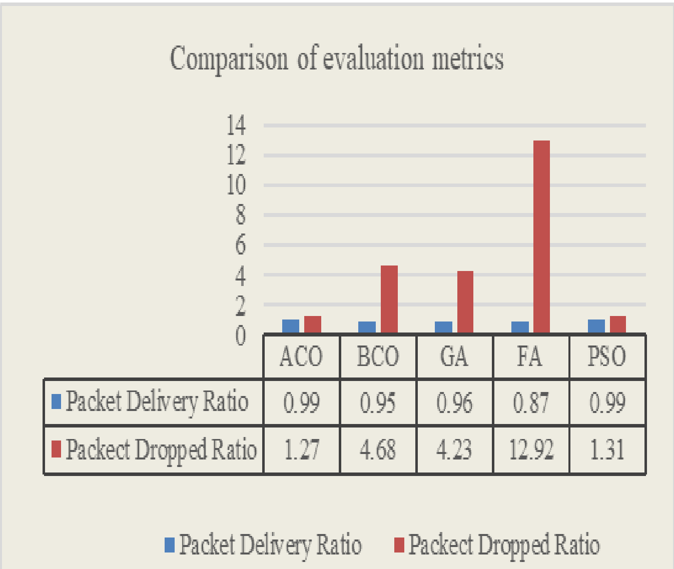

Figure 1 Packet Delivery Ratio and Packet Dropped Ratio

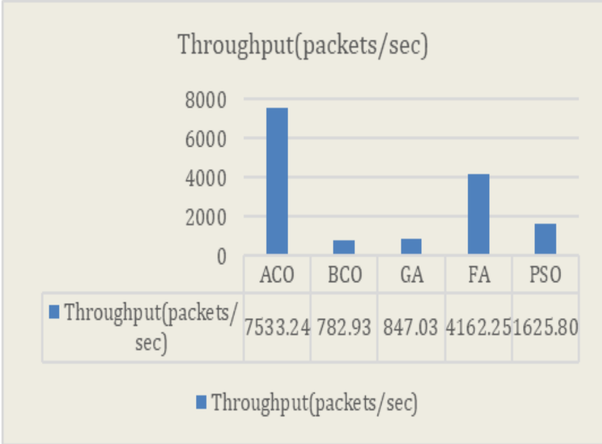

Figure 2 Throughput

By imitating ant foraging behavior, ACO has demonstrated potential in solving combinatorial optimization challenges. Traditional ACO implementations, however, are less effective in real-world applications due to issues including high [4] end-to-end delay and fault vulnerability. Timely decision-making can be hindered by end-to-end delay, or the amount of time it takes to identify a solution, and system reliability can be jeopardized by fault vulnerability. Figure 3 depicts the high End to End delay in ACO.

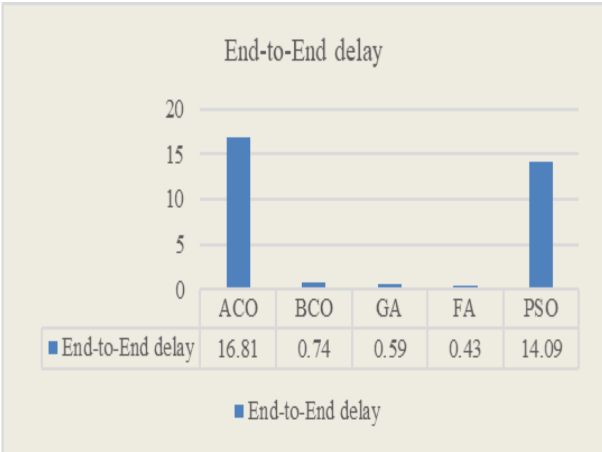

Figure 3 End-To-End Delay

1.4 Introduction to Machine Learning (ML)

Machine learning (ML) techniques (Sun, Y., et al., 2021;) have revolutionized the field of optimization, providing effective tools to improve efficiency and effectiveness. In this project, we use ML methods to improve the [5] performance of

IRJAEH
International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

ACO for combinatorial optimizations. In particular, we use a Random Forest model to predict survival, a SVM model to predict failure, and a LASSO model to [6] select features. Random forest model is used to predict system survival, which allows the ACO to anticipate and adapt to potential disruptions. On the other hand, the SVM model is used to predict failures, which enables the ACO to proactively repair failures and maintain system reliability. Additionally, [7] LASSO is used for feature selection, which helps identify and prioritize relevant parameters for optimization. By integrating these ML techniques into ACO, we aim to improve its performance, reduce endpoint latency, and improve system reliability.

2. Literature Review

In order to further enhance the performance of traditional ACO, several existing methods were surveyed. The enhancement of ACO with ML techniques were preferred to bring an unique solution to our high end to end delay problem. Feature selection was employed here (Zhou J. et al., 2008;) along with fault prediction in ACO. This paper is very much similar to the proposed methodology but this solution is used for machinery tool wear out application. The following [8] solution discusses supervised feature selection method based on ant colony optimization for software fault detection prediction. It also employs KNN, Naive Bayes and Decision Tree classifiers. Another approach of enhancing ACO included integration of ML with ACO in context of solution prediction As mentioned in (Mondal S. et al.,2023;) the objective is to find a route that visits a subset of vertices in a graph within a time budget to maximize the collected score. This solution was used to solve combinatorial optimization problems such as orienteering problem.

3. Methodology

A comprehensive methodology that integrates machine learning (ML) techniques using Matlab to improve the performance of Ant Colony Optimization (ACO) in solving combinatorial optimization issues. Our three different machine learning models are as follows: [9] the Random Forest model predicts survival, the Support Vector Machine (SVM) model predicts faults, and the Least Absolute Shrinkage and Selection Operator (LASSO) model selects features.

3.1 Ant Colony Optimization (Aco) Algorithm

Ant Colony Optimization (ACO) (Nayar N., et al., 2021;) is a metaheuristic inspired by the behavior of real ants. [10] ACO is employed to facilitate fault prediction and adaptation within the Flying Ad-hoc Network (FANET) environment (Beegum T. R., et al., 2023;).

Initialization: Initialization involves setting up parameters such as the number of ants, pheromone levels, and heuristic information. Pheromone trails are initialized on network paths, representing the desirability of those paths based on historical experience or heuristic knowledge.

Ant Movement: Ants traverse the network environment, making decisions on their movement based on both pheromone levels and heuristic information (Wang S., et al., 2016). Each ant probabilistically selects the next node to visit, favoring paths with higher pheromone concentrations and shorter distances.

Pheromone Update: After all ants have completed their tours, pheromone levels on network paths are updated based on the quality of solutions found. Paths with better performance (e.g., lower fault occurrence) receive higher pheromone deposits, reinforcing their attractiveness for future exploration.

3.2 Integration of ML Techniques with ACO

The Random Forest model is utilized to predict system survivability, enabling ACO to anticipate potential disruptions and adapt its optimization strategy accordingly. The SVM model is employed for fault prediction, allowing ACO to proactively identify and mitigate faults, thus enhancing system reliability. Moreover, [11] LASSO is employed for feature selection, aiding in the identification and prioritization of relevant parameters for optimization. Each ML model operates independently, with separate feature selection processes tailored to optimize the performance of the corresponding prediction task. This comprehensive approach ensures that ACO can effectively leverage the predictive capabilities of ML techniques while maintaining its and

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

efficiency in solving optimization problems.

Random Forest Model: During training, the Random Forest model builds many decision trees and produces the mode of the classes (classification) or the average prediction (regression) of the individual trees (Jackins, V., et al.,2021;). This technique is widely used in ensemble learning. Predicting survivor-ship using the Random Forest model. In order to forecast the chance of system survival under various circumstances or scenarios, this entails training the model using historical data. The model fits this purpose well since it can handle complex data and capture nonlinear relationships (Zheng J., et al.,2020;).

SVM Model: For tasks involving regression and classification, the Support Vector Machine (SVM) is a potent supervised learning technique. Within the feature space, SVM (Farooqui., et al., 2020) looks for the hyperplane that best divides classes. SVM model for fault prediction. This involves training the model on labeled data to classify instances as either faulty or non-faulty based on features extracted from the system. SVM's ability to handle high-dimensional data and effectively classify instances with clear margins makes it well-suited for fault prediction tasks (Zhang Y. -s., et al., 2022;)

Lasso: For feature selection and regularization in linear regression models, the regularization approach known as LASSO (Least Absolute Shrinkage and Selection Operator) is employed. It penalizes the regression coefficients absolute sizes, leading to sparse solutions where irrelevant features are set to zero (Muthukrishnan R., et al., 2016;) By utilizing LASSO for feature selection. This involves applying LASSO regularization to identify and prioritize relevant features that [12] contribute most significantly to the prediction tasks performed by the Random Forest and SVM models. By selecting only the most relevant features, LASSO (Kumarage P. M., et ai., 2019;) helps streamline the optimization process and improve model performance.

3.3 Performance Metrics

Packet Delivery Ratio (PDR): Proportion of transmitted packets that reach the destination without loss. High PDR indicates efficient data delivery.

Packet Dropped Ratio: Percentage of packets that fail to reach the destination. High drop ratio impacts system performance negatively.

Average End-To-End Delay: The duration of a packet's journey from its origin to its final destination. Reliability in real-time applications depends on little delay.

Throughput: Rate at which data flows through the network. High throughput ensures efficient data delivery.

Overall Survivability: Network's resilience to failures, attacks, or adverse conditions. High survivability indicates a robust network.

4. Simulation Setup

The simulation environment is configured for consistency, integrating the Ant Colony Optimization (ACO) algorithm into the FANET framework. Relevant network and environmental data are gathered. [13] The key features used to train the ML models are as follows:

UAVs positions: It defines the position of each UAVs by X and Y coordinates.

Pheromone Levels: It represents the concentrations of pheromone trails deposited by UAVs within the ACO framework. It guides UAV movement decisions, communication routing, and fault prediction exploration.

Path Quality: It characterizes the suitability and reliability of UAV trajectories within the FANET. It considers factors such as terrain conditions, obstacle density, and signal propagation effects.

Battery Levels: It denotes the remaining energy levels of individual UAVs' batteries.

Survivability: It is considered as a network's resilience to failures, attacks, or adverse conditions.

Fault Label: It is a binary variable which stores data on whether a faulty node is detected or any type of failure occurs during the communication between UAVs. Random Forest models for survivability (Kumar, S., et al., 2023) and fault occurrence are trained using selected features. Predictions for target variables are made using these models. The ACO strategy is updated based on these predictions, ensuring its responsiveness to changing conditions. In Figure 4. the flow of the project which represent the methodology of the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

project

Figure 4 **Flow hart**

These parameters collectively shape the operational context of the simulated FANET environment and serve as foundational elements for evaluating the performance and effectiveness of the ACO and ML-ACO frameworks in fault prediction and adaptation using Matlab.

5. Results and Discussion

The primary goal is to address the challenges of high end-to-end delay and susceptibility to faults in traditional ACO implementations. By leveraging ML methods, particularly the Random Forest model for fault prediction and [14] LASSO feature selection for optimization, significant improvements in system performance and survivability are anticipated. The following section evaluates the performance of the proposed ML-enhanced ACO approach based on key performance metrics.

5.1 Performance Metrics Evaluation

Packet Delivery Ratio (PDR): The Packet Delivery Ratio (PDR) achieved in the experiment was an impressive 0.99, indicating a high success rate in delivering packets to their intended destinations.

Packet Dropped Ratio: The Packet Dropped Ratio observed was 1.45, signifying a relatively low occurrence of packet losses within the network infrastructure.

Average End-to-End Delay: The Average End-to-End Delay recorded was 6.15 seconds, reflecting the time taken for data packets to traverse the network from source to destination.

Throughput: The Throughput measured in the experiment amounted to 7574.69 packets/second, demonstrating the network's efficiency in transmitting data within a given timeframe.

Overall Survivability: The Overall Survivability of the system was determined to be 95.38%, indicating its robustness and resilience to faults and disruptions. This result shows that the ML-ACO outperforms traditional ACO by enhancing survivability, minimizing delay, and maintaining efficient data transfer rates.

Survivability (expressed in Figure 5), Throughput (expressed in Figure 6), and End to End Delay (expressed in Figure 7) provide a comparative analysis on ACO and ML-ACO.

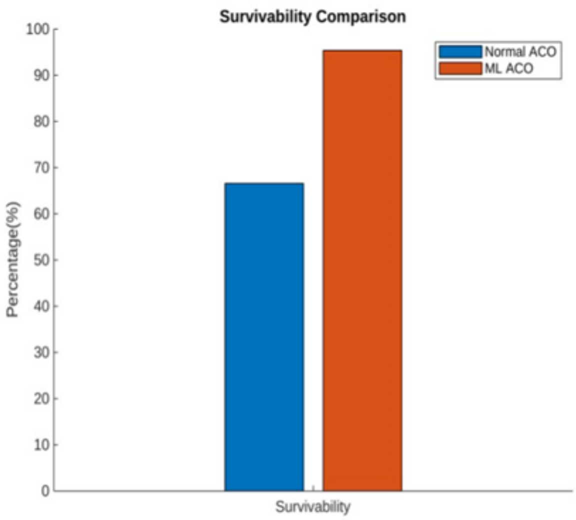

Figure 5 **Simulated Graph of Survivability**

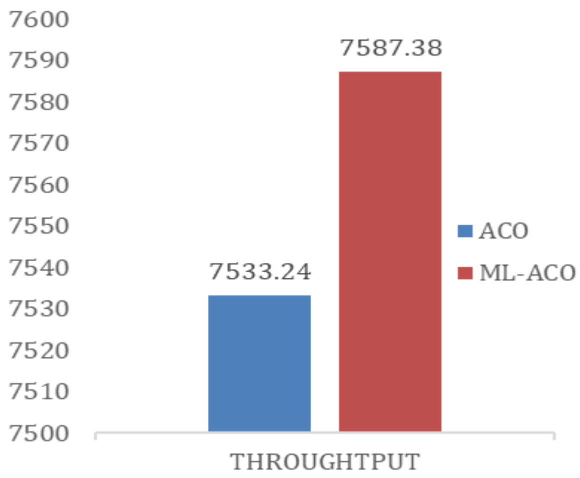

Figure 6 **Simulated Graph of Throughput**

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 02 Issue: 05 May 2024
Page No: 1200 - 1206
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2024.0165

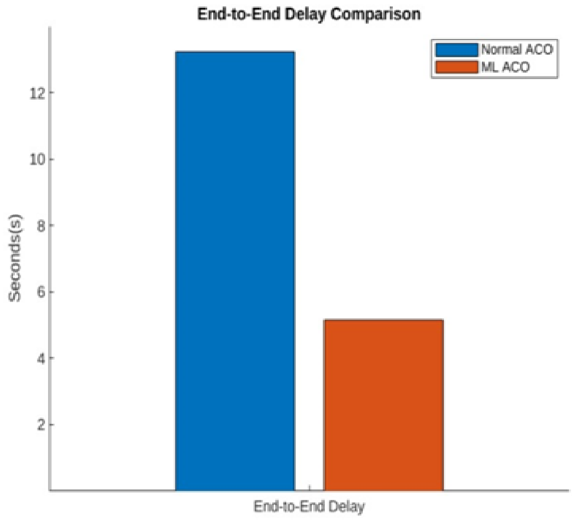

Figure 7 Simulated Graph of End-To-End Delay

Table 1 Comparison of ACO And ML-ACO

ALGORITHMS	SURVIVABILITY(%)	END TO END DELAY(s)	THROUGHPUT (Packets/sec)
ACO	66.667	13.23	7544.5
ML-ACO	95.38	5.15	7574.6

Table 1 effectively show the improvement of ML integrated ACO when compared with the traditional ACO. The values of survivability, End to End Delay, [15] and throughput are significantly improved thus the expected outcome is reached.

5.2 Discussion

The obtained results highlight the effectiveness of the proposed ML-enhanced ACO approach in addressing the challenges of high end-to-end delay and susceptibility to faults in traditional ACO implementations. The integration of machine learning techniques, particularly the Random Forest model for fault prediction and LASSO feature selection for optimization, has led to significant improvements in system performance and survivability. These findings underscore the potential of the proposed approach for real-world applications in optimization problems, offering promising avenues for further research and development.

Conclusion

In Conclusion, The implementation of ML techniques has demonstrated significant improvements in the ACO algorithm. More efficient and effective results are achieved compared to traditional ACO. This enhancement highlights the potential of combining ACO with ML for optimizing FANET performance, showcasing a promising direction for future research and development in this field.

References

[1]. Zhou J., Ng R. and Li X., "Ant colony optimization and mutual information hybrid algorithms for feature subset selection in equipment fault diagnosis," 2008 10th International Conference on Control, Automation, Robotics and Vision, Hanoi, 2008, pp. 898-903, doi: 10.1109/ICARCV.2008.4795637.

[2]. Mondal S., Sahu A. K. , Kumar H., Pattanayak R. M., Gourisaria M. K. and Das H. , "Software Fault Prediction using Wrapper based Ant Colony Optimization Algorithm for Feature Selection," 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1-6, doi: 10.1109/ISCON57294.2023.10111995.

[3]. Sun Y., Wang S., Shen Y., Li X., Ernst A.T., Kirley M., "Boosting Ant Colony Optimization via Solution Prediction and Machine Learning" https://doi.org/10.48550/arXiv.2008.04213.

[4]. Zhang Y. -s., Ming F. and Chang M. -j. , "A prediction model for slope stability based on the support vector machine," 2022 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), Shijiazhuang, China, 2022, pp. 41-45, doi: 10.1109/ICCEAI55464.2022.00018.

[5]. Zheng J., Xin D., Cheng D., Tian M., Yang Le., The Random Forest Model for Analyzing and Forecasting the US Stock Market in the Context of Smart Finance 2020, https://doi.org/10.48550/arXiv.2008.04213

[6]. Farooqui, Ehtisham Md. and Ahmad, Dr.

Jameel, Disease Prediction System using Support Vector Machine and Multilinear Regression (August 13, 2020). International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN: 2347-5552, Volume, 8, Issue, 4, July, 2020, Available at SSRN: https://ssrn.com/abstract=3673232 or http://dx.doi.org/10.2139/ssrn.3673232

[7]. Muthukrishnan R. and Rohini R. , "LASSO: A feature selection technique in predictive modeling for machine learning," 2016 IEEE International Conference on Advances in Computer Applications (ICACA), Coimbatore, India, 2016, pp. 18-20, doi: 10.1109/ICACA.2016.7887916.

[8]. Kumarage P. M., Yogarajah B. and Ratnarajah N., "Efficient Feature Selection for Prediction of Diabetic Using LASSO," 2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2019, pp. 1-7, doi: 10.1109/ICTer48817.2019.9023720.

[9]. Beegum T. R. , Idris M. Y. I., Ayub M. N. B. and Shehadeh H. A. , "Optimized Routing of UAVs Using Bio-Inspired Algorithm in FANET: A Systematic Review," in IEEE Access, vol. 11, pp. 15588-15622, 2023, doi: 10.1109/ACCESS.2023.3244067.

[10]. Jackins, V., Vimal, S., Kaliappan, M. et al. AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes. J Supercomput 77, 5198–5219 (2021). https://doi.org/10.1007/s11227-020-03481-x

[11]. Sun, Y., Ernst, A., Li, X. et al. Generalization of machine learning for problem reduction: a case study on travelling salesman problems. OR Spectrum 43, 607–633 (2021). https://doi.org/10.1007/s00291-020-00604-x

[12]. Wang W., Zhang J., Wang S., Chen X., “Effective fault module localization in substation critical equipment: an improved ant colony optimization and back propagation neural network approach” October 2023, https://doi.org/10.1049/tje2.12315

[13]. Kumar, S., Rathore, N.K., Prajapati, M. et al. SF-GoeR: an emergency information dissemination routing in flying Ad-hoc network to support healthcare monitoring. J Ambient Intell Human Comput 14, 9343–9353 (2023). https://doi.org/10.1007/s12652-022-04434-3

[14]. Wang, S., Song, W. Experimental Study of Ant Movement in a Straight Passageway under Stress Conditions. J Insect Behav 29, 735–743 (2016). https://doi.org/10.1007/s10905-016-9593-x

[15]. Nayar, N., Gautam, S., Singh, P., Mehta, G. (2021). Ant Colony Optimization: A Review of Literature and Application in Feature Selection. In: Smys, S., Balas, V.E., Kamel, K.A., Lafata, P. (eds) Inventive Computation and Information Technologies. Lecture Notes in Networks and Systems, vol 173. Springer, Singapore. https://doi.org/10.1007/978-981-33-4305-4_22