Evolutionary Intelligence (2025) 18:47
https://doi.org/10.1007/s12065-025-01032-y

RESEARCH PAPER

Neural network-driven scenario prediction for adaptive routing in MANETs using expanding ring search and random early detection

M. A. Gunavathie[1] · Ujwal Ramesh Shirode[2] · Nichenametla Rajesh[3] · V. Sudha[4]

Received: 26 August 2024 / Revised: 10 February 2025 / Accepted: 1 March 2025 / Published online: 27 March 2025

Abstract

Routing in Mobile Ad Hoc Networks (MANETs) is complex due to their decentralized topology and dynamic environments. Conventional routing protocols frequently have trouble maintaining reliable end-to-end communication, leading to issues such as high packet loss, unstable routes, and increased delays. The research presented a novel deep-learning model with hybrid optimization to maintain route stability through stable node prediction and optimal route discovery in a MANET environment. Here Dynamic Sparse Recurrent-Convolutional Neural Network (DSR-CNN), an advanced variant of the R-CNN family is employed to detect the stable node in a sparse and dynamic environment. The weight parameters of the DSR-CNN are optimized by the Enhanced Fick's Law Optimization Algorithm for the improved node prediction maintaining reliability and connectivity. Moreover, the most stable routes are constructed through the Giant Trevally Optimizer, optimizing route exploration and recovery. The model is designed and simulated in a Python environment with the required network settings. The utilization of the proposed research in the MANET communication shows 98% prediction accuracy, 2.48 ms delay, 25Mbps throughput, and 99% of packet delivery ratio which are more efficient than the prevailing technique's results showing the highest route stability. These findings highlight the potential of DSR-CNN to significantly improve the reliability and performance of MANETs, setting a new benchmark for future routing protocols.

Keywords Mobile Ad Hoc networks (MANETs) · Routing prediction · Expanding ring search (ERS) · Random early detection (RED) · Dynamic network topology

1 Introduction

A MANET is a set of nodes that are mobile-linked by wireless networks working as a single system. Indeed, it has no permanent equipment attached to it. Each node of the network is a router to the next node that is in the next higher layer of the network [1, 2]. Some interesting characteristics of MANET are described below. In case a node migrates from one location to another, for instance, MANET can be flexible, portable, and link devices. By identification of route, information packets can be forwarded from the node of origin to the neighboring node and eventually to the target node [3, 6]. Among the traditional routing methods, AODV and DSR are prevalent in dealing with network modifications and path determination. However in order to optimize these protocols one has to be equally familiar with the settings and the network parameters where these protocols are to be implemented.

The use of routing standards is important in Mobile Ad Hoc Networks (MANETs) for efficient communication. The

✉ M. A. Gunavathie
gunavathie.ap@gmail.com

Ujwal Ramesh Shirode
ujwalshirode@gmail.com

Nichenametla Rajesh
nrajeshcse@kluniversity.in

V. Sudha
sudha.ece@sonatech.ac.in

1 Department of Information Technology, Easwari Engineering College, Chennai, Tamil Nadu, India

2 Department of Electronics and Telecommunication (E&TC), Pimpri Chinchwad College of Engineering (PCCoE), Pune, Maharashtra, India

3 Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India

4 Department of Electronics and Communication Engineering, Sona College of Technology, Salem, Tamil Nadu, India

decision of choosing the right route protocol and its various setting options for a given network topology can be considered one of the most important tasks that define the network's behavior, at the same time [7–9]. One of the suggestions that have been made to address the routing challenges in MANETs is machine learning (ML). ML algorithms can use data-driven models to make routing decisions based on past experience but choose different routings given the current status of the network. The main advantage of machine learning in network routing is its ability to make superior and adaptive routing choices based on real-time data and network conditions. Various machine learning methods, such as supervised, unsupervised, and reinforced learning have recently been studied with a view of achieving the best route in MANETs. They have also shown the ability to enhance network reliability, reduce power consumption, and optimize the routing aspect [10–12].

Many routing protocols used in MANETs employ such approaches as extending ring search (ERS), random early detection (RED), and numerous additional addresses and variations, successfully using a number of routing parameters across various formats to meet the QoS and address such problem s as congested, caused by water damages, energy drain, expenses, and link damages. Employs a neural network to predict the optimal conditions under which ERS and RED parameters are to be deployed in MANETs. These forecasts have to do with network properties such as topology, node mobility, and surrounding conditions. The overall goal is to optimize the network metrics by adapting the routing strategies with reference to the state of the network with regard to PDR, throughput, and end-to-end latency [13–15]. However, the typical process of using neural networks for routing entails fashioning a neural network design suitable for routing activities, feeding the neural network with updated data concerning the relevant networks, and then incorporating the resultant neural networks-based routing scheme into the existing routing systems. Despite the weakness of this method like the increased computational cost and the necessity of consistent reference training data, it opens more flexibility and efficiency in the field of the current network management that provides the viability of this field in future research and development. It is proposed, therefore, that the Dynamic Sparse R-CNN is used. The outcomes of the proposed research have significant practical implications for enhancing the efficiency, stability, and security of MANET applications in real-world scenarios. MANETs are utilized in various critical applications, such as disaster recovery, military operations, vehicular networks, and IoT-based smart environments, where the absence of a fixed infrastructure requires robust, adaptive communication protocols. The researched model employed the mechanisms of Dynamic Sparse R-CNN for stable node prediction and Enhanced Fick's Law Optimization for route stability, which can greatly improve the overall performance of MANETs.

1.1 Novelty and contributions

- This study proposes a novel approach for predicting stable routing paths in MANETs using a combination of advanced neural network techniques and optimization algorithms.
- Using Dynamic Sparse R-CNN, stable nodes are predicted.
- There is control over the expert weights of the nodes using certain softmax techniques. However, enhanced Fick's Law Optimization is applied for efficient outcomes.
- Then, the employment of the Giant Trevally Optimizer improves the Routing process by selecting the optimal for the source-to-destination data travel.

The remaining of this manuscript is prearranged as Sect. 2 reviews the literature; Sect. 3 suggests a methodology; Sect. 4 presents the findings and discussions; and Sect. 5 concludes.

2 Literature survey

Zafar and Altalbe. [16] proposed machine-learning techniques to enhance network performance by using parameter regression to determine the optimal protocol and routing parameters. The network's infrastructure was trained to predict network throughput, end-to-end (E2E) latency, and packet delivery ratio (PDR). Simulations were performed in different network topologies to assess network operation. The model harnessed the strengths of ML architecture to upgrade parameters of routing algorithms such as ERS and RED, paying more attention to optimizing key performance metrics such as throughput, E2E latency, and PDR.

Abbood et al. [17] developed a machine-learning technique using software-defined networks (SDNs) to improve mobile ad hoc network (MANET) performance by distributing node burden more fairly and extending the wireless sensor network (WSN) lifespan. It discovers alternative packet transmission pathways, decreasing reliance on low-energy nodes and increasing hops per packet.

Duong. [18] created an enhanced Ad hoc On-Demand Distance Vector (AODV) protocol for 5G MANETs using reinforcement learning to maintain a state data database that increases throughput, reduces end-to-end delay, and improves signal-to-noise ratio (SNR) efficiency.

Baird et al. [19] introduced a method for detecting AODV routing protocol attacks in MANETs. The method used signature verification using the Andrews digital signed curve to guarantee secure communication between nodes. The OMNet + + emulator proved this method by detecting between 83 and 100% of malicious nodes. The security improvement enhanced the efficiency of the network, minimizing packet dropping and maximizing throughput, which is a key factor in reliable MANET communication.

Shafi et al. [20] proposed a Machine Learning and Trust-Based AODV Routing Protocol (ML-AODV) to combat flooding assaults in MANETs. It selects cooperative intermediary nodes using trust estimation, thereby decreasing packet drop rates. The protocol identifies suitable paths using machine learning techniques such as Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The ML-AODV increases the throughput and reliability up to 4% as well as the latency, routing overhead, and packet loss ratio by 12%, 15%, and 10%, respectively. Similar to the proposed research, ML-AODV highlights the improvement in network performance as well as network reliability by identifying appropriate nodes to route the communication.

Militani et al. [21] proposed the e-RLRP protocol, which applies reinforcement learning to decrease overhead by adjusting the 'Hello' communication interval dynamically to counterbalance additional latency. The network characteristics were simulated and VoIP quality was predicted using the E-model technique in experiments. The e-RLRP protocol efficiently minimizes routing overhead, thereby reducing unnecessary communication between nodes and consequently lower bandwidth consumption.

Jothi Lakshmi and Karishma. [22] proposed the implementation of the DSR protocol for wireless MANETs with the concept of deep reinforcement learning. This protocol is said to improve network efficiency through the integration of scheduling algorithms and congestion control methods by enhancing traffic oversight, connection servicing, and channel access control. The employment of DRL in the DSR protocol allows it to optimize decisions in the area of routing concerning learned traffic patterns and known network conditions that may help select the optimal transmission paths for data.

Wheeb and Naser. [23] performed an assessment of active, receptive, and numerous path routing protocols in MANETs using simulation to observe the performance behavior of AOMDV, DSDV, and AODV protocols as a function of the mobile node counts, pause rate, and various traffic connectivity parameters.

Wheeb and Al-Jamali. [24] tested and evaluated the performance of the Optimized Link State Routing (OLSR) protocol over various mobility models, namely Random Waypoint (RWP), Random Direction (RD), Nomadic Community (NC), and Reference Point Group Model (RPGM). The findings showed that OLSR is superior to the RWP model in reliability, data loss ratio, latency, and routing complexity in low-mobility environments.

Wheeb and Kanellopoulos. [25] highlighted the importance of Quality of Service (QoS) support for multimedia streaming applications over MANETs using TCP-friendly rate control (TFRC) and stream control transmission protocol (SCTP). The study compares SCTP and TFRC in terms of important QoS metrics such as throughput, end-to-end latency, and PDR.

Mekkaoui et al. [26] proposed energy-aware and link-stability routing protocol known as ES-RPDE, evolved strategy routing protocol with differential evolution. This protocol selects optimal paths through a fitness function that uses energy consumption and link stability to determine the path being used.

Pandey and Singh. [27] proposed Stable-Ad hoc On-Demand Distance Vector protocol to enhance the stability of route and improve performance of AODV routing. It chooses a next hop that has maximum residual node energy as well as higher received signal strength compared to dynamically threshold levels. Gnanasekaran et al. [28] have come up with Optoelectronic Intelligent Stable Routing Protocol (OISRP), by integrating optoelectronic devices with the already existing system of wireless networks, to enhance signal quality detection. The study compared OISRP with AODV in terms of round-trip latency, PDR, and route lifespan.

ul Hassan et al. [29] presented a smart cluster-based routing model to prevent black hole attacks in VANETs. The system uses an ANN model to detect malicious nodes and constructs clusters with specified cluster heads. The approach integrates an improved On-demand route finding and appropriate path selection using the AODV protocol. Table 1 shows the summary of the reviewed Approach.

2.1 Problem statement

The primary problem in the MANET routing is maintaining the stable and efficient route due to the continuous changes of the nodes position and environment. Nodes travel autonomously and at random based on their speed of movement. Frequent node mobility leads to the route breakage. This increases the packet-drop rate in the routing path and causes connection problems. Packet delivery may be paused or lost due to unreliable nodes. As a result, the upgraded routing cannot be built. Current MANET routing protocols struggle with maintaining connectivity and low latency due to frequent topology changes. The proposed DSR-CNN model tackles these challenges by accurately predicting stable nodes and optimizing routing paths, thereby reducing packet loss and improving overall network performance. Stable nodes are predicted using Dynamic Sparse R-CNN in order to get over these problems. The enhanced Fick's Law Optimization procedure is applied to improve the Stable nodes

Table 1 Summary of the reviewed approach

References	Methods	Objectives	Limitations
Zafar and Altalbe. [16]	ERS, RED	Improve network performance using ML-based parameter optimization	Ensuring high prediction accuracy is challenging in dynamic networks
Abbood et al. [17]	DRL	Balance workload distribution and extend WSN lifespan	Training DRL models requires significant computational resources
Duong. [18]	AODV	Enhance 5G MANET routing with RL-based state maintenance	Increased computational demands on nodes
Baird et al. [19]	ML-AODV	Detect and prevent malicious AODV attacks	Cryptographic operations introduce computational overhead
Shafi et al. [20]	ANN-SVM	Select trusted relay forwarders using ML-based trust estimation	High computational complexity
Militani et al. [21]	RL	Reduce routing overhead using adaptive Hello intervals	Requires careful RL parameter tuning
Jothi Lakshmi and Karishma. [22]	DRL	Optimize DSR with congestion control and scheduling	High system complexity
Mekkaoui et al. [26]	ES-RPDE	Optimize routing based on energy and link stability	Does not consider latency or throughput
Pandey and Singh. [27]	SAODDV	Improve route stability in AODV	Dynamic thresholds may cause routing oscillations
Gnanasekaran et al. [28]	OISRP	Enhance routing with optoelectronics	Hardware and software integration challenges
ul Hassan et al. [29]	SC-ANN	Detect malicious nodes and form optimal clusters	May not detect sophisticated black hole attacks

by providing precise weight values. Next, the recommended optimizer known as the Giant Trevally Optimizer is used to process the performance metrics of the improved routing in order to establish it.

The motivation for using Recurrent Convolutional Neural Network (RCNN) techniques arises from their unique capability to combine the strengths of both convolutional layers (CNNs) and recurrent layers (RNNs). This hybrid structure allows RCNNs to effectively capture spatial patterns from input data through convolution, while also modeling sequential or temporal dependencies using recurrence, making them particularly useful for applications involving dynamic and time-varying information. In the context of MANETs, this capability is highly advantageous due to the dynamic nature of network topologies and the need for both spatial and temporal insights in routing decisions. Some research has employed RCNNs in ITS integrated VANET to predict traffic flow and vehicle movement. These models integrate spatial features from road networks with temporal traffic patterns to optimize routing decisions, which shares similarities with MANET routing problems.

3 Proposed methodology

In this proposed methodology, the stable node and its respective Routing path prediction follow three stages: predicting the stable node, stability quantity determination, and route examination procedure. As shown in Fig. 1, the workflow diagram in this study represents the proposed DSR-CNN model that considers stability prediction, optimization, and route recovery mechanisms for the routing process in MANETs. The process initiates with stable node prediction using dynamic sparse recurrent-convolutional neural networks, where stable nodes are identified according to the mobility pattern and link quality. The stable nodes found are further put through Stability Evaluation Using Enhanced Fick's Law Optimization so that only the stable nodes are chosen for routing. After stability evaluation, Route Establishment with the Giant Trevally Optimizer phase optimizes path selection to improve reliability and reduce communication overhead. Neighboring Stable Node Data Gathering and Routing enables effective data transmission by dynamically adapting to network conditions. If a route is not available or needs optimization, the Routing Exploration and Recovery mechanism searches for an alternative path in order to not break the communication. Finally, the model goes through Performance Evaluation and Analysis of important performance indicators such as packet delivery ratio, delay, and throughput. The holistic approach guarantees that MANET routing remains stable, efficient, and resilient to dynamic topology changes.

In MANETs, nodes are constantly moving, which causes frequent topology changes. Fick's Law, when applied, models node stability based on diffusion principles. If nodes move too frequently, the diffusion model may struggle to predict reliable connections, leading to an increase in route breaks and packet loss. As mobility increases, node stability

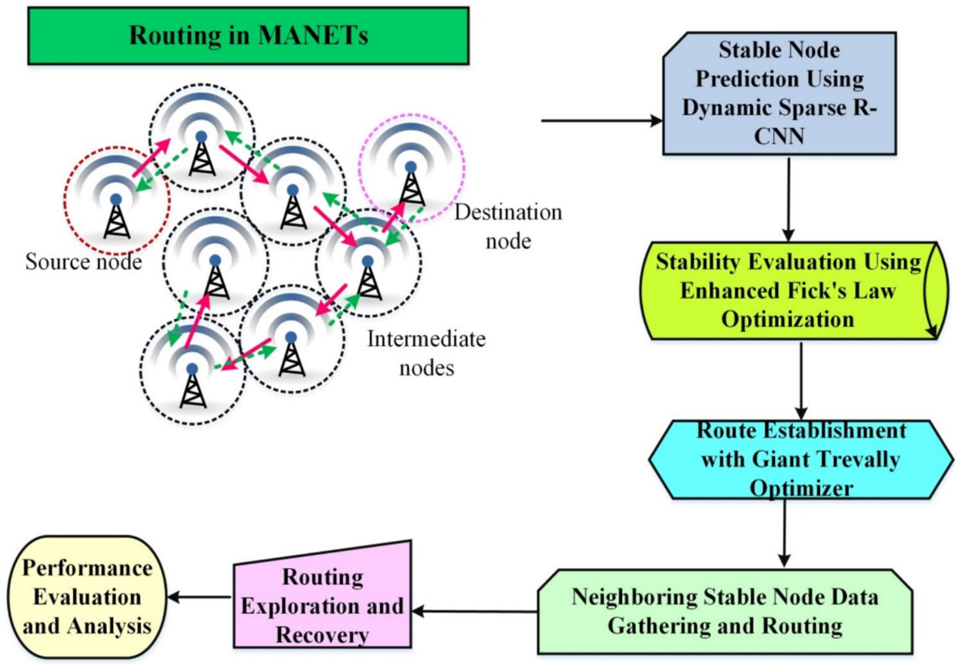

Fig. 1 Workflow diagram of DSR-CNN

decreases, reducing the effectiveness of Fick's Law in maintaining stable routes. The law assumes a relatively consistent diffusion of stability across nodes, but with high mobility, the network experiences uneven diffusion, making route predictions less reliable. The diffusion coefficient governs how quickly the "stability" or "reliability" information diffuses across the network. In a highly dynamic MANET environment, adjusting the diffusion coefficient based on node mobility can help better reflect the actual network conditions. In addition to the diffusion coefficient, the parameters such as node stability gradient, mobility-aware adjustment factor, TTL for stability data, diffusion range, and node density consideration of Fick's Law can be adjusted to make the model more effective for handling node mobility in MANETs. These adjustments ensure that the law reflects the dynamic nature of MANETs and provides more reliable node stability and route recovery predictions, ultimately improving network performance.

3.1 Dynamic sparse R-CNN for stable node prediction

Many of the existing approaches are worked out to anticipate the stable nodes. It failed because of some differences in its features. The Dynamic Sparse R-CNN [30] approach is used to get around the restrictions. A member of the R-CNN family of models, dynamic sparse R-CNN is intended to maximize object detection and routing in environments with sparse and dynamic data. Convolutional neural networks (CNNs) are utilized for the recognition of objects and localization, and routing choices are modified in real time based on network conditions and the stability of nodes' importance [31]. For stable node prediction, lively sparse R-CNN can strengthen the robust sparse R-CNN starting point with various mainstays The DSR-CNN is highly effective for stable node prediction in MANETs due to several specific attributes of its architecture and function. In Mobile Ad Hoc Networks (MANETs), node stability is inherently temporal. Nodes may be stable for some time before moving or becoming unreliable. The recurrent component of the DSR-CNN excels in capturing temporal dependencies by retaining information about past states, making it ideal for predicting node behavior over time. By learning the history of a node's movements and connection patterns, DSR-CNN can predict future stability more accurately than models that only consider the present state.

To predict the stable node for the routing scenario, the neural taken utilizes the different network-based input features such as Node Mobility, Node Energy Levels, link quality, hop count, node density, link duration, traffic load, and node connectivity degree to predict the stability. Expanding Ring Search (ERS) is a widely used technique in MANETs to discover routes between source and destination nodes while minimizing network flooding and overhead. ERS works by initially sending route request (RREQ) packets within a small time-to-live (TTL) radius. If a route to the destination is not found within this radius, the search is repeated with an incrementally larger TTL value, thus gradually expanding the search area. If a route is not discovered within the maximum TTL, the process fails, prompting either another method of search or network reconfiguration. Scenario prediction through the neural network enhances the performance of ERS by improving its efficiency in dynamically changing network environments. With input features like node energy, mobility, and link duration, the neural network can predict which nodes are likely to form stable routes. ERS can then prioritize expanding the search toward regions with more stable nodes, improving the likelihood of finding reliable routes quickly.

3.1.1 Dynamic sparse R-CNN architecture

Each iterative step receives three types of input: proposal features, suggestion containers, and FPN features that are retrieved by the backbone. Predicted boxes, matching classes, and article characteristics of the boxes are included in the output. The following step uses the revised suggestion boxes and object features that the previous phase's anticipated boxes and object characteristics produced are employed as. A tiny fixed set of area proposals that show the possible positions of the items are called proposal boxes. Latent vectors used to encode instance properties, such as form and posture, are known as proposal features. Proposal boxes in Dynamic Sparse R-CNN are fixed for inference after being learned during training. In order to create a two-part corresponding amongst forecasts and crushed truth substances, Dynamic Sparse R-CNN implements the set-based damage. This matching method combines one-to-one correspondence with the Hungarian algorithm to ensure accuracy. Dynamic Sparse R-CNN is an enhancement of the Sparse R-CNN framework, primarily aimed at addressing the limitations of the assignment of labels one to one method used in traditional object detection models. The main reason for implementing dynamic set-based approaches in R-CNN is that the assignment of labels one to one scheme, which uses the Hungarian algorithm, is often not optimal for matching trained proposal boxes with ground truths. This can lead to inefficiencies in training and subpar performance in object detection tasks, the dynamic set-based approach in Dynamic Sparse R-CNN is implemented through DLA and DPG, which together enhance the model's ability to adaptively generate and assign proposals, leading to improved performance in node detection task.

3.1.2 Dynamic label assignment

The Hungarian method is applied in Dynamic Sparse R-CNN for one-to-one correspondence, in which every pulverized truth is coordinated to a single prophesied box. The Hungarian method is applied to assign each predicted bounding box to exactly one ground truth object. This method ensures that every ground truth object is matched with one and only one identified box, thus avoiding redundant or incorrect assignments. The process anticipates that this kind of one-to-one correspondence won't be ideal. In some cases, forcing a one-to-one match might cause certain prediction boxes to remain unassigned, even if they could partially correspond to ground truth objects. This can result in missed opportunities for training and may negatively impact performance in scenarios where multiple predicted boxes could validly represent parts of an object or capture different aspects of ground truth instances. Numerous positives assigned to a GT can improve prediction training and optimize the proposals more effectively. Prediction boxes are treated as demanders in the formulation, seeking assignments, whereas ground facts are treated as suppliers providing quota for assignment. Additionally, the backdrop class is designed to function as a provider of default assignment. Let's take a mathematical example where it have p ground truths in a node and each of them gives us assignments or units. Every q prediction box aims to become a unit, and a positive match is known as a positive project match. Negative assignments, or prediction containers that are not allocated to any crushed truth, are fulfilled by the background using units. The optimization objective is formulated in Eq. (1),

$$
\begin{aligned}
&\min_{\omega} \sum_{i=1}^{p} \sum_{j=1}^{q} K(i,j) * \omega(i,j),\\
&s.t. \sum_{i=1}^{p} \omega(i,j) = 1,\\
&\sum_{j=1}^{q} \omega(i,j) = \delta_i,\\
&\sum_{i=1}^{p} \delta_i = q,\\
&\omega(i,j) > 0, i = 1,2,...,p, j = 1,2,...,q,\\
&K(i,j) = \begin{cases} L_{\text{class}}(i,j) + \beta * L_{\text{regression}}(i,j), & \text{positive assignment} \\ L_{\text{class}}(\text{background}, j), & \text{negative assignment} \end{cases}
\end{aligned}
\tag{1}
$$

where i is the directory of crushed truth, j is the directory of prediction containers ($j = 1, ...q$), $K(i,j)$ refers to a cost or loss associated with assigning i to j, $\omega(i,j)$ characterizes the similar effect to be optimized amongst ground actuality i and prediction box j, β is an amount harmonising the arrangement and regression sufferers, L_{class} is a classification loss, $L_{regression}$ is a regression loss, δ_i represents the number of assignments, r is the numeral of items. This equation represents the optimization objective for dynamic label assignment, where the goal is to suppress the total assignment cost.

Each confident assignment has a cost equal to the total of the regression and classification losses, whereas each negative assignment has a cost of only the classification losses. Each provider may supply a fixed or variable quantity of units. Determine the r-value, or the relationship between the estimates and the ground-truth cartons, dynamically using the Dynamic Estimation technique. The estimation for the r value in this method is calculated by adding the top t IoU values for each. In general, the Dynamic Estimation technique is valid. Supposing that p is the number of GTs and A_x is the number of total suggestions, if $p \times r > 80\% \times A_x$, we will decrease r by a similar scaling issue for both GTs to safeguard at smallest 20% negative projects.

3.1.3 Unit increasing strategy

Iterative architecture is used by Dynamic Sparse R-CNN to progressively improve prediction accuracy. To encourage the training of an iterative structure, a straightforward unit-increasing technique is proposed. It supposes the providers GT supply a limited quantity of units when the active bonce forecasts are not accurately abundant in the premature stages, which forces the matching to be more stringent. In the latter stages, as the dynamic head forecasts grow more accurate, we slowly loosen the restrictions to allow the dealers (GT) to supply more components for equivalent. The unpretentious unit growing approach is given in the Eq. (2),

$$r^* = r - 0.5 * (S - s), s = 1, 2, ...S \tag{2}$$

where r^* represents the adjusted value of r based on the current time s, r is the initial value of r, S represents the total number of time steps or periods over which the strategy is applied, and s represents the current time step or period. The avoidance amount of iteration periods (S = 6) is used in our method. In the early stages of training, r^* is reduced more strictly, which ensures that the matching process is more stringent and limits the number of positive assignments. As training progresses, r^* is adjusted to allow more flexibility, with the final goal of achieving better prediction accuracy by providing more units for assignments.

3.2 Dynamic proposal generation

In dynamic Sparse R-CNN, a set of A_x application containers and A_x application characteristics together with the characteristics, that is sent into the dynamic head. taken out from the FPN spine (P_2 to P_5). these applications are learnable throughout working out but immobile for dissimilar nodes throughout implication. The proposed model generates dynamic suggestion boxes and features with admiration to the contribution node to recover presentation. The subsequent diagram signifies the dynamic suggestion cohort. Suggestion boxes/features are a linear mixture of B_y separate sets of suggestion boxes/features, and both sets are mentioned as *expert*. The constants (also known as expert weights) to interconnect the specialists are produced by a competent weights cohort network. DPG unit is characterized in appearance as follows (Eq. 3):

$$\begin{aligned} U_o^g &= \sum\nolimits_{i=1}^{B_y} U_i^g * V_i \\ U_o^h &= \sum\nolimits_{i=1}^{B_y} U_i^h * V_i \\ (V_1,\ V_2, \ldots, V_{B_y}) &= W(H) \end{aligned} \tag{3}$$

where U_i^g indicates the output lively suggestion boxes, U_i^h expresses the output lively suggestion features, V_i is the application expert weight scholarly by the practiced weight generation network W, H specifies the features pull out from the FPN backbone (P2 to P5). Using certain operations in softmax, the expert weights of the nodes are controlled to make the training process effective. Some fluctuations are seen in predicting the stable nodes. To rectify this issue, an Improved Fick's Law optimization algorithm is used.

3.3 Fick's law optimization

To enhance the prediction accuracy of the DSR-CNN, the research integrated enhanced Fick's law optimization. The important parameters of the DSR-CNN such as weight, learning rate, bath size, drop-out rate, and regularization influence the stable node prediction accuracy. These parameters are optimally tuned by the enhanced Fick's law optimizes to attain the maximum prediction performance. "Fick's law optimization" [32] is an algorithmic approach that applies principles analogous to Fick's law of diffusion to solve optimization or simulation problems. It involves iteratively adjusting parameters to optimize an objective function or simulate diffusion-like processes in various domains. EFL's global search capability helps in finding the optimal solutions across the entire search space which is important for the most stable route recovery in the complex MANET environment. The Fick's Law Algorithm is an influential optimization technique that can fairly predict the stable nodes with accurate weight parameters by the following steps:

4 *Step 1:* Initialization

In the enhanced Fick's law algorithm, optimization starts based on the weight parameters of Dynamic Sparse R-CNN gathered while accurately predicting its node stability.

5 *Step 2:* Fitness function

The objective function, which is to correctly predict the stable node without any fluctuations in weight parameters and to increase the performance accuracy, is done by using the fitness features. The fitness features for enhancing the performance accuracy by reducing the weight parameter of established network expression is shown in Eq. (4).

$$\text{Fitness Function} = \text{Max}(\text{Performace Accuracy})/\text{Min}(V_i) \quad (4)$$

6 *Step 3:* Update molecular position

The three phases proposed to perform the fitness function include the Dispersion Phase, Evenness Phase, and Steady Government Phase. To execute between three phases, the following equation is used (Eq. 5):

$$U_j^q = \begin{cases} DP\ T_f^q < .9 \\ EP\ T_f^q \le 1 \\ SSP\ T_f^q > 1 \end{cases} \quad (5)$$

6.1 Diffusion phase (DP)

Initially, the dual districts have a tall variation in attentiveness, this clues to transmission particles from one district to additional rendering to the attentiveness of the assumed region. The proposed stricture is given by:

$$T_{DP}^q = F_5 \times T_f^q - c \quad (6)$$

$$F_5 = 2 \quad (7)$$

On the basis of T_{DP}^q, the course of flow is strong-minded as:

$$U_{s,j}^q = \begin{cases} \text{from } j \text{ to} i \text{ region } T_{DP}^q < c \text{ and} \\ \text{from } i \text{ to } i \text{ region otherwise} \end{cases} \quad (8)$$

For i's particles, they alter their location in a similar area without itinerant as i region is advanced in attentiveness so they inform their location rendering to the symmetry location in region i and the border of the problematic rummage-sale can be intended from the subsequent equation:

$$U_{s,i}^{q+1} = U_{EP,i}^q + dof \times (c_4 \times (H - S) + S \quad (9)$$

where F_5 is the scaling factor, $U_{s,j}^q$ is the symmetry location in the area j, $U_{s,i}^{q+1}$ is the symmetry location in the region I, T_f^q is a transfer function with respect to time q, c is a constant, dof refers to the diffusion of flow, H refers to the higher concentration and S refers to the lower concentration. The overhead approach will be measured if section j is minor in concentration and if region i is higher, then the vice versa will occur, where the molecules travel from i's region to j's region.

6.2 Equilibrium phase (EP)

From exploration to exploitation, the equilibrium phase is seen as a transitional stage. The molecules use Eq. (10) for the position updation,

$$U_{s,h}^{q+1} = U_{EP,i}^q + P_{EP,h}^q \times U_{s,h}^q + P_{EP,h}^q \times (NT_{s,EP}^q \times U_{EP,h}^q - U_{s,h}^q) \quad (10)$$

$NT_{s,EP}^q$ is calculated by the Eq. (11),

$$NT_{s,EP}^q = e\left(-\frac{PA_{h,EP}^q}{(PA_{s,h}^q + EPS)}\right) \textit{motion step} \quad (11)$$

where $U_{s,h}^{q+1}$ is the location of subdivision in collection h, $P_{EP,h}^q$ mentions to the comparative number of the district in the assembly h, $NT_{s,EP}^q$ refers to motion step, $PA_{h,EP}^q$ refers to the greatest qualification groove in assemblage at stretch q and $PA_{s,h}^q$ is the qualification score of particle s in collection at stretch q.

6.3 Steady state phase (SSP)

The concluding point in the optimization examination is the manipulation or permanence phase and the,

$$NT_{s,h}^q = e\left(-\frac{PA_{SSP}^q}{(PA_{s,h}^q + EPS)}\right) \quad (12)$$

The accurate fitness score is processed by all three phases and the value is executed (Eq. 12).

Table 2 Weight parameter values

Layer	Filter/Node	Weights
First convolutional layer	Filter 1	0.12, – 0.23,0.450.12, -0.23, 0.450.12, – 0.23,0.45, 0.11,0.36, – 0.480.11, 0.36, -0.480.11,0.36, – 0.48, – 0.19,0.22,0.31–0.19, 0.22, 0.31 – 0.19,0.22,0.31
	Filter 2	0.09, – 0.32,0.240.09, -0.32, 0.240.09, – 0.32,0.24, 0.15,0.29, – 0.410.15, 0.29, -0.410.15,0.29, – 0.41, – 0.08,0.19,0.34–0.08, 0.19, 0.34 – 0.08,0.19,0.34
Second convolutional layer	Filter 1	0.05, – 0.18,0.27,0.08, – 0.120.05, -0.18, 0.27, 0.08, -0.120.05, – 0.18,0.27,0.08, – 0.12, 0.14, – 0.35,0.42,0.03, – 0.210.14, -0.35, 0.42, 0.03, -0.210.14, – 0.35,0.42,0.03, – 0.21
	Filter 2	0.07, – 0.26,0.33, – 0.15,0.110.07, -0.26, 0.33, -0.15, 0.110.07, – 0.26,0.33, – 0.15,0.11, 0.02, – 0.19,0.28, – 0.09,0.060.02, -0.19, 0.28, -0.09, 0.060.02, – 0.19,0.28, – 0.09,0.06
Fully connected layer	Node 1	0.21, – 0.45,0.33,0.18,0.070.21, -0.45, 0.33, 0.18, 0.070.21, – 0.45,0.33,0.18,0.07
	Node 2	– 0.12,0.29, – 0.37,0.22,0.05–0.12, 0.29, -0.37, 0.22, 0.05 – 0.12,0.29, – 0.37,0.22,0.05

7 *Step 1:* Termination

Improved Fick's law Optimization algorithm is terminated by finding the optimized output of a stable node. The obtained weight parameter values of the proposed DSR-CNN are detailed in Table 2.

The present research calculates the weight parameters of the DSR-CNN with a hybrid optimization approach. It optimizes the model's weight values using Enhanced Fick's Law Optimization Algorithm for fine-tuning the parameters of the network for stable node prediction in MANET. This technique optimizes weights according to the behavior of the network under dynamic and sparse conditions in such a manner that the weights make proper predictions of the stable nodes. Initialization of the weights occurs followed by iterative updates during the training process to reduce prediction error for nodes and improve the network in terms of reliable communication routes. These optimized weight values, as shown in Table 2, are the final set of parameters learned by the network to achieve improved routing stability and connectivity in MANETs.

7.1 Determination of stable neighboring nodes

In order to broadcast packets from the foundation to the terminus, the appropriate routing is built after the forecast of subsequent stable nodes. In the event that the Source and Destination maintain the same interval, neighboring nodes or substitutes are not required. However, it is required to locate some neighboring nodes in order to construct an alternate routing path in the event that the stable node is unable to transport packets because of a lag in its parameters. Neighbour stable nodes are predicted using the enhanced Fick's Law optimization technique. The recommended Giant Trevally Optimizer is then utilized to determine an optimal route.

7.2 Routing exploration

The route exploration scheme known as the Giant Trevally Optimization establishes a navigation path in the secure nodes. Packets are transmitted to the intended location by the nearby nodes. The route reply packet leaves to node that is the target to the source node once it has received the packets that were sent. As a result, the source nodes calculate each path that relies on the stability value. The routing of nearby stable nodes is confirmed by the Routing Exploration Scheme. In the event that a routing link breaks, route recovery will be employed.

7.3 Giant trevally optimizer (GTO)

The Giant Trevally Optimizer (GTO) was specifically chosen for routing stability in Mobile Ad Hoc Networks (MANETs) due to its unique attributes and performance advantages over traditional optimization algorithms. GTO is inspired by the foraging behavior of the Giant Trevally fish, which is known for its efficiency in locating prey in dynamic environments. This biological analogy allows GTO to adaptively search for optimal solutions, making it well-suited for the constantly changing conditions of MANETs**.** The Giant trevally Optimizer [33] is inspired by nature and gets derived for the implementation of the applications mathematically. Being a top predator, the giant trevally hunts with intelligence. It is known that the gigantic trevally hunts both alone and in groups. Predators who hunt in groups are the most successful at taking down their prey. The leader, or first predator, is the group member who is best at catching prey. The design of the GTO was primarily influenced by these cutting-edge hunting techniques, which include foraging movement patterns, selecting the right location in terms of food amount, and leaping purchasable of the liquid to assault and capture the prey.

8 *Step 1:* Initialization

GTO initiates the optimization course by randomly producing initialization explanations called giant trevallies.

9 *Step 2:* Fitness function

In this period, rummaging movement outlines of giant trevallies are imitation using the Fitness function.

$$Fitness \;\; function = (Maximum(t_p.P_{dr})Avg(s))/Minimum(j.e) \tag{13}$$

where t_p denotes the throughput, P_{dr} is the ratio for packet deliveries, s indicates the node velocity, j represents the jitter and e is End 2 End delay.

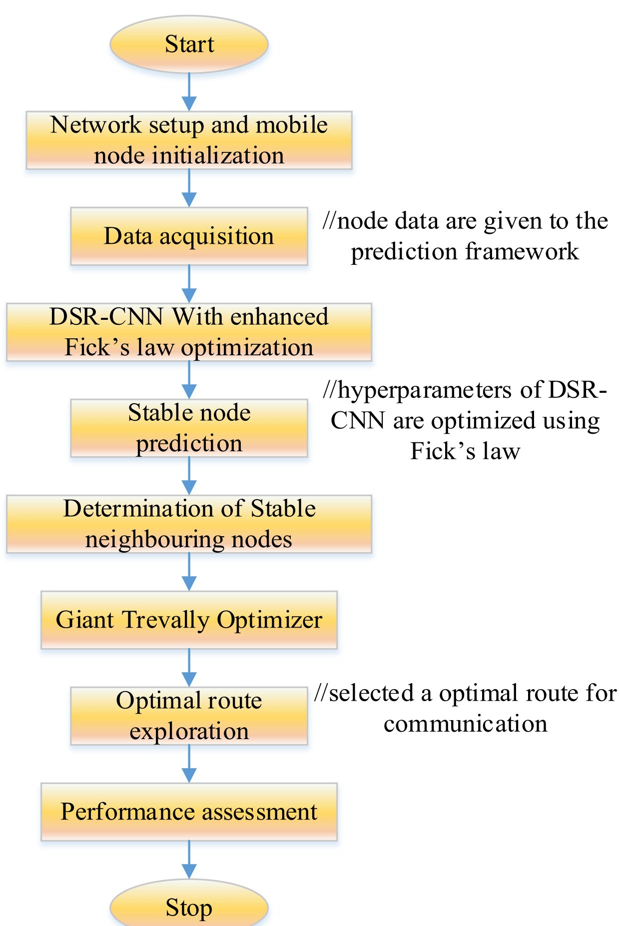

Fig. 2 Flowchart of proposed method

10 *Step 3:* Attacking

The behaviour of huge trevally Optimizer while racing and hopping out of the water is simulated exactly using the Eq. (14),

$$R(s+1) = \ell + \vartheta + \chi \tag{14}$$

where $R(s+1)$ is the resolution of the ensuing iteration of s, ℓ is the Launch speed, ϑ signifies the visual distortion, χ is the jumping slope function.

11 *Step 4:* Termination

After the establishment of enhanced routing by iterating the attacking process, iteration is stopped and the testing process and terminated. The process of the proposed research for the stable routing is defined in Fig. 2.

12 Results and discussion

A Python and NS-3 combination is used to execute the simulation procedure for evaluating this proposed DSR-CNN model. In fact, the actual implementation of the DSR-CNN model with Python is meant to train the deep learning network for stable node prediction and the optimal discovery of routes. Libraries such as TensorFlow or Keras may have been used to construct and train the convolutional neural network, while other Python libraries like NumPy and SciPy were used for mathematical computations and optimization tasks, such as fine-tuning the weight parameters through

Table 3 Simulation parameters

Parameters	Description/values
Simulation platform	Python
Routing protocol	DSR
Area	1000*1000m^2
Nodes	120
Simulation time	2000s
Bandwidth	10Mbps
Transmission range	250 m
Node speed	2- 50 m/sec
Pause time	0 to 900 s
Packet generation rate	2–6 packets/sec
Packet size	512 bytes
No. of connections	5
Traffic type	CBR

the Enhanced Fick's Law Optimization Algorithm. On the other hand, NS-3 simulated the MANET environment: it created its own dynamic characters for mobile nodes, packet transmission, and route discovery processes. The proposed optimizations to enhance the routing in MANETs are examined based on its performance metrics using Giant Trevally Optimizer. The results are tabulated in the following subsections. The network environmental settings for implementing the proposed research are detailed in Table 3.

In the current research, the time taken for simulation was chosen as 2000s or approximately 33.1 min with consideration of the balance between computational feasibility and inclusion of meaningful performance metrics of MANETs. With a simulation time of 2000s, typically the connectivity stability between nodes, the effectiveness of routes to be discovered, and the efficacy of the routing protocol driving various network conditions should be adequately comprehended. It supports a set of concurrent events such as node mobility, packet transmission, route discovery, and recovery and captures the complete system behaviour without adding unnecessary computational overhead. The prediction and routing process of the proposed research is implemented with the link stability database (LSD). In the created network settings, all the nodes are placed with the LSD that stores the connections and information pertaining to nodes for the prediction of the stable node dynamically and also establishes a stable route to broadcast the packets to the destinations. The LSD maintains the information of the parameters such as distance, energy level, antenna-related data, and node ID, stability factor, and bit error rates.

12.1 Performance estimation

Efficient routing is based on performance metrics which include Node velocity, Jitter, End to End delay, throughput, packet drop, and PDR. The significant parametrs to determine the optimized routing is described below.

12.1.1 Node velocity

In node velocity, the upper speed limits up to which the node can move. Actual tall or actual low speeds tend to damagingly show effects on routing.

12.1.2 Jitter

Jitter is the term used to refer to the variance in packet arrival times caused by network congestion or altered routes. Retaining minimal jitter facilitates effective routing.

12.1.3 End-to-end delay (E2E delay)

E2E latency is the measure of duration that it takes for a packet to travel across a network between its initial and final nodes. Featured are all of the delays, including those related to inquiry processing, finding routes, buffering, retransmission queuing, development, and transfer. After measuring in nanoseconds, the delay is translated to seconds. This can be computed using the formula found in (15),

$$E2E_{average} = \sum_{n=0}^{m} \left(\frac{n^{th} \text{ Packet received time} - n^{th} \text{ Packet Sent time}}{\text{Number of Packets received in total}} \right) \tag{15}$$

12.1.4 Packets drop

In this instance, packets that are correctly transferred to the target node during the communication operation and are not missed are referred to.

12.1.5 Packet delivery ratio

Packet Delivery Ratio is a phrase used to describe the measurement of consecutive packet delivery from origin to recipient. The effectiveness of the routing protocol is also gauged by this parameter. The packet delivery ratio is calculated using (16), which has a mathematically illustrated formula found in (16),

$$\text{Packet Delivery Ratio} = \text{received packets}/\text{Transferred Packets} * 100 \tag{16}$$

12.1.6 Throughput

The data transmission rate between source and destination nodes at a specific time is represented by throughput. Below is a performance study of various metrics: Fig. 3 displays the Throughput, Jitter, and Packet Delivery Ratio are analyzed across different numbers of nodes in a network. The proposed DSR-CNN method is compared with four existing methods, AODV, ERS, ANN, and DRL. AODV and ERS are well-established, widely-used routing protocols in MANETs. These protocols are frequently used as baselines because they represent standard solutions for on-demand and reactive routing in dynamic networks. Even though they don't utilize deep learning, they provide a clear reference point for evaluating whether advanced methods (like deep learning) actually bring a measurable improvement in performance. By comparing with AODV and ERS, it can demonstrate how the proposed deep learning-based model can outperform traditional routing protocols, particularly in dynamic and

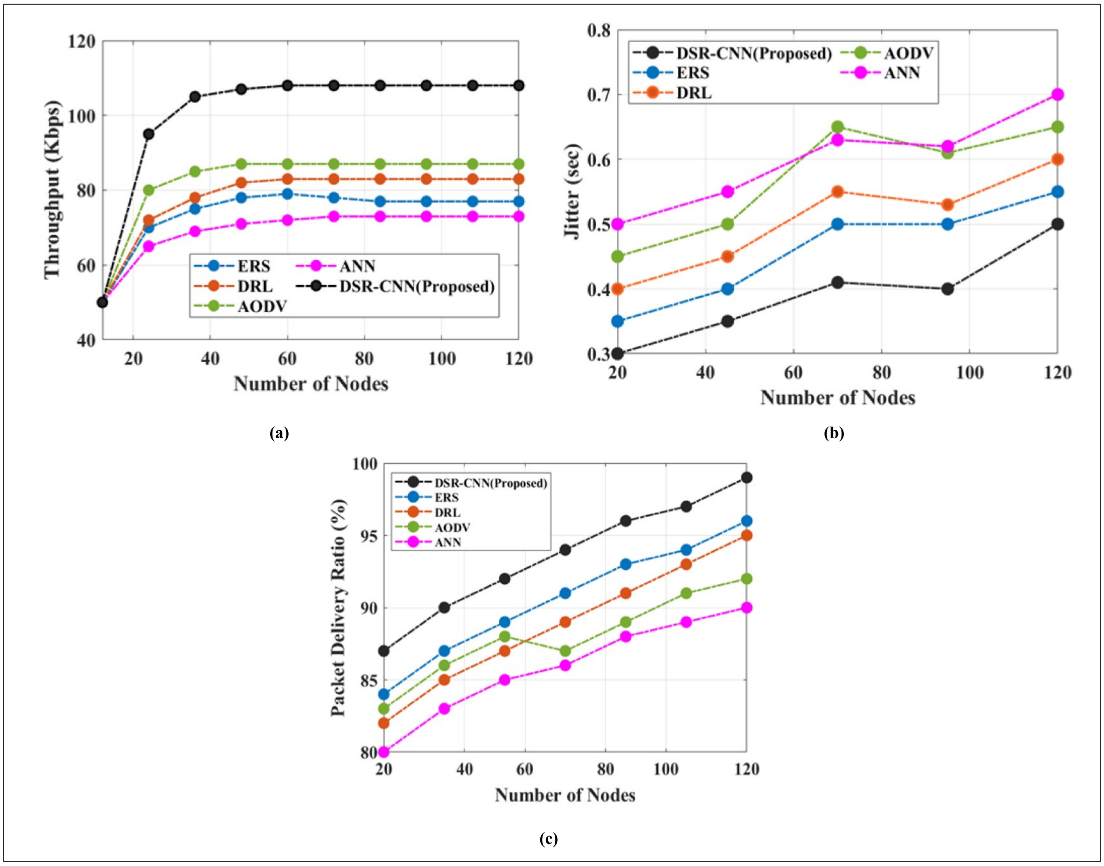

Fig. 3 **a** Throughput, **b** jitter, and **c** packet delivery ratio

sparse environments where the limitations of AODV and ERS (like route discovery latency, packet loss, and instability in high-mobility conditions) become prominent.

Figure 3a displays the throughput is measured in Kbps as a function of the nodes count. The proposed DSR-CNN algorithm shows the highest throughput consistently across all node counts, peaking at around 120 Kbps as the number of

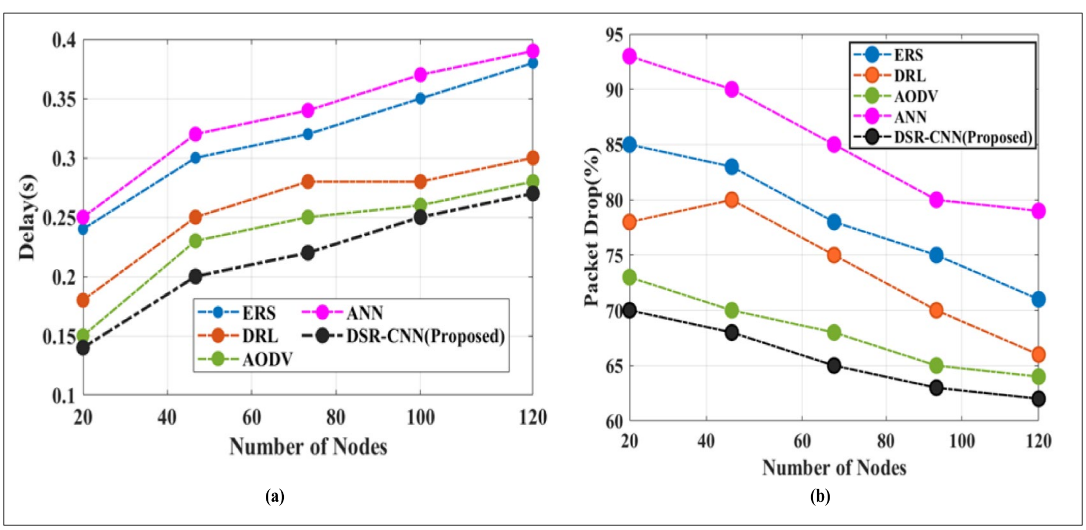

Fig. 4 **a** Delay and **b** Packet Drop Rate nodes increases, indicating superior data transfer efficiency. The other existing algorithms exhibit lower throughput, with AODV performing the poorest, particularly at higher node counts. Figure 3b displays the jitter versus the number of nodes. Jitter represents the variation in packet arrival time, and lower values are preferable. The DSR-CNN algorithm demonstrates the lowest jitter, especially as the number of nodes increases, signifying more stable and reliable communication. In contrast, ANN shows the highest jitter, which worsens with an increasing number of nodes, suggesting instability under higher network loads. Figure 3c shows the PDR as a percentage relative to the number of nodes. The approach termed as DSR-CNN eclipses all the other approaches in terms of PDR, emergent across the different nodes and nearly hitting 100% as the network is scaled. This means, that compared to regular fiber networks, the proposed algorithm can help to reduce less successful packet deliveries. The rest of the algorithms – namely, ANN and AODV – have notably worse PDR, with ANN being the worst-performing algorithm in each category. A high PDR ensures that most of the data sent across the network reaches its destination, which is critical for ensuring communication reliability. In MANETs, where nodes are mobile and network topologies constantly change, maintaining a high PDR helps ensure that messages are not lost due to unstable or broken routes.With a higher PDR, network resources such as bandwidth and energy are used more effectively because more packets reach their destinations without the need for additional resource allocation (e.g., retransmission). This is vital in environments like MANETs where network resources are often limited.

Figure 4 shows the Delay and Packet Drop Rate analyzed across various numbers of nodes in a network. The proposed DSR-CNN method is compared with four existing methods, AODV, ERS, ANN, and DRL. Figure 4a demonstrates the delay, measured in seconds, and is plotted against the number of nodes. The proposed DSR-CNN algorithm consistently gets the smallest delay for all number of nodes, indicating faster data transmission and processing times. As the amount of bulges grows, the delay for ANN and ERS increases significantly, making them less efficient in larger networks. ANN, in particular, exhibits the highest delay, suggesting that it struggles to maintain performance as the network scales. Figure 4b shows the packet drop rate as a percentage in relation to the number of nodes. Here, the proposed DSR-CNN approach demonstrates the lowest packet drop rate, showing its effectiveness in maintaining data integrity as the network size grows. The ANN algorithm, on the other hand, has the highest packet drop rate, which worsens with an increasing number of nodes, indicating that it is less reliable in ensuring that data packets reach their intended destination. The other algorithms, such as DRL and AODV, fall between these two extremes but still show higher packet drop rates compared to the proposed DSR-CNN model. Minimizing delay and packet drop ratio is essential in MANETs to ensure efficient, reliable, and timely communication. Low delay allows for real-time applications, smooth communication, and better resource utilization, while a low packet drop ratio guarantees data integrity, reduces congestion, and enhances overall network performance. The combination of these improvements makes the proposed routing protocol highly effective for dynamic environments like MANETs, where stability, efficiency, and real-time communication are crucial.

Table 4 provides detailed specifications of End-to-End Delay data used in experiments, comparing different routing

Table 4 Specifications of end to end delay data used for experiments

Parameters	Methods	Dmin	Dmax	Davg	Dsdv
Speed	AODV	18.37	17.45	17.84	1.397
	ERS	19.58	13.26	16.88	2.188
	ANN	15.46	16.34	16.12	1.374
	DRL	11.26	13.27	12.86	1.386
	DSR-CNN (Proposed)	6.37	8.74	7.92	1.275
Terrain	AODV	2.81	14.37	8.46	4.83
	ERS	2.16	15.28	9.37	2.47
	ANN	2.35	8.49	6.92	3.75
	DRL	2.85	7.94	5.28	5.37
	DSR-CNN (Proposed)	1.63	3.28	2.48	2.17
Node density	AODV	3.28	5.38	4.86	6.83
	ERS	2.47	7.37	4.92	5.38
	ANN	5.38	5.37	5.23	6.37
	DRL	3.57	6.36	4.93	4.96
	DSR-CNN (Proposed)	2.13	3.65	2.95	3.95

Table 5 Specifications of throughput data used for experiments

Parameters	Methods	Tmin	Tmax	Tavg	Tsdv
Speed	AODV	14.25	15.47	15.12	2.475
	ERS	16.47	12.28	14.65	3.285
	ANN	12.74	13.37	12.98	1.975
	DRL	13.53	12.26	12.75	1.864
	DSR-CNN (Proposed)	18.37	19.64	17.98	1.627
Terrain	AODV	2.26	13.28	9.74	7.387
	ERS	2.74	12.84	8.39	7.265
	ANN	2.93	9.47	6.39	8.375
	DRL	2.96	8.85	5.93	8.972
	DSR-CNN (Proposed)	3.84	13.85	10.26	7.163
Node density	AODV	5.94	40.27	26.48	10.38
	ERS	5.17	39.72	24.85	11.38
	ANN	5.28	36.73	22.48	13.83
	DRL	5.39	35.82	24.92	14.28
	DSR-CNN (Proposed)	6.97	45.28	28.48	16.83

methods: AODV, ERS, ANN, DRL, and the proposed DSR-CNN approach are compared under 3 different criteria: Speed, Terrain, and Node Density. As for the Speed parameter, for the DSR-CNN method, the minimum average delay is a Davg of 7. 92 s, and the least standard deviation (Dsdv) of 1 s. 275, which proves this trend to be constant and effective. In Terrain, DSR-CNN once more produces the lowest average delay of 2.48 s and a standard deviation of 2.17, putting it in a position to perform well in different terrains. In Node Density, DSR-CNN has the lowest average delay of 2.95s and lowest standard deviation of 3. 95 this shows the efficiency of DSR CNN in a high node density. Generally, for all parameters, it is observed that DSR-CNN is the best as it provides the lowest delay along with the least variation in its values as compared to the other approaches that have been discussed above.

The Table 5 presents the specifications of throughput data used in experiments, comparing five routing methods, AODV, ERS, ANN, DRL, and the proposed DSR-CNN across three different parameters: Speed, Terrain, and Node Density. The DSR-CNN method consistently demonstrates superior performance with the highest average throughput (Tavg) across all parameters. For Speed, DSR-CNN achieves the highest average throughput at 17.98 with a relatively low standard deviation (Tsdv) of 1.627, indicating stable performance. In Terrain, DSR-CNN also leads with a Tavg of 10.26, outperforming other methods that show higher variability and lower averages. Under Node Density, DSR-CNN again excels with the highest average throughput of 28.48 and a Tsdv of 16.83, suggesting that it maintains higher data transmission rates even as the network density increases. Overall, the DSR-CNN (Proposed) method proves to be the most efficient and reliable in terms of throughput, outperforming other methods across various network conditions.

The Table 6 provides the specifications of Packet Delivery Ratio (PDR) data used in experiments, comparing various routing methods—AODV, ERS, ANN, DRL, and the proposed DSR-CNN—across three parameters: Speed, Terrain, and Node Density. The DSR-CNN method consistently achieves the highest average PDR (Davg) across all conditions, indicating its superior ability to deliver packets successfully within the network. For Speed, DSR-CNN has an impressive average PDR of 94.86%, with a low standard

Table 6 Specifications of packet delivery ratio data used for experiments

Parameters	Methods	Dmin	Dmax	Davg	Dsdv
Speed	AODV	71.37	89.37	83.74	8.3868
	ERS	68.84	92.48	91.65	6.4665
	ANN	73.58	91.38	78.75	7.8624
	DRL	75.38	93.56	82.64	4.8767
	DSR-CNN (Proposed)	87.45	98.73	94.86	1.3874
Terrain	AODV	78.86	99.53	85.38	7.375
	ERS	82.78	99.46	91.37	6.387
	ANN	84.75	99.93	93.47	7.486
	DRL	83.65	99.12	92.47	5.386
	DSR-CNN (Proposed)	86.86	99.75	94.57	2.746
Node density	AODV	78.48	84.74	83.87	1.987
	ERS	79.84	85.76	88.94	2.975
	ANN	82.47	83.47	87.38	2.354
	DRL	81.35	86.78	86.37	1.366
	DSR-CNN (Proposed)	88.36	92.56	91.36	3.576

Table 7 Least RMSE and MAE values

Methods	Parameters	Node density		Terrain		Speed	
		RMSE	MAE	RMSE	MAE	RMSE	MAE
AODV	Delay	0.162	0.173	0.263	0.274	0.249	0.115
	PDR	0.073	0.057	0.074	0.057	0.286	0.264
	Throughput	0.054	0.084	0.076	0.081	0.187	0.165
ERS	Delay	0.026	0.072	0.274	0.176	0.186	0.127
	PDR	0.274	0.164	0.084	0.075	0.083	0.042
	Throughput	0.026	0.065	0.038	0.064	0.145	0.076
ANN	Delay	0.186	0.169	0.027	0.174	0.176	0.145
	PDR	0.026	0.075	0.076	0.093	0.086	0.025
	Throughput	0.176	0.185	0.157	0.135	0.186	0.175
DRL	Delay	0.026	0.074	0.076	0.169	0.165	0.081
	PDR	0.038	0.076	0.081	0.186	0.169	0.027
	Throughput	0.189	0.084	0.076	0.174	0.274	0.076
DSR-CNN (Proposed)	Delay	0.013	0.064	0.083	0.042	0.029	0.049
	PDR	0.072	0.037	0.063	0.072	0.082	0.026
	Throughput	0.024	0.017	0.019	0.015	0.028	0.035

Table 8 Comparison with state of art models

Methods	Accuracy	MAE	RMSE
ERS [16]	85%	0.12	0.15
DRL [17]	90%	0.09	0.12
AODV [18]	78%	0.15	0.2
ML-AODV [19]	88%	0.1	0.13
ANN-SVM [20]	92%	0.08	0.11
RL [21]	87%	0.11	0.14
DRL [22]	93%	0.07	0.1
Proposed	98%	0.06	0.09

Table 11 t-test statistical results

Methods	*t*-statistic	*p*-value
LR	177.1	4.83E-14
KNN	168.39	6.87E-14
FCNN	168.76	6.77E-14
CNN	79	1.37E-11
RNN	66.54	4.55E-11
Proposed	-	-

deviation (Dsdv) of 1.3874, signifying stable and reliable performance. In Terrain, DSR-CNN also leads with an average PDR of 94.57% and a moderate standard deviation of 2.746, outperforming other methods that exhibit higher variability and lower averages. Regarding Node Density, DSR-CNN continues to excel, achieving the highest average PDR of 91.36%, reflecting its robustness in dense network environments. Overall, the DSR-CNN (Proposed) method demonstrates exceptional performance in maintaining a high packet delivery ratio, outperforming other methods under various network conditions.

The above Table 7 shows the Least RMSE and MAE of the proposed method are better than the existing method. In all the metrics and conditions, it has been found that the Proposed DSR-CNN has the lowest RMSE and MAE through the DSR-CNN method. In general, it yields the least RMSE for Delay (0. 013), PDR (0. 072), and Throughput (0. 024) for different node densities, terrain profiles, and velocities, which confirms the effectiveness of the present model in predicting these parameters. Likewise, for Delay, it provides the least MAE values of 0. 064, for PDR it is 0.037, and Throughput is 0. 017 which asserts the fact of low variance. The low RMSE and MAE of the DSR-CNN approach in all conditions bring out the fact that DSR-CNN is capable of reducing prediction errors to the lowest level.

The overall comparison with the state-of-the-art methods regarding the prediction metrics such as accuracy, RMSE, and MAE for assessment of the prediction ability with the literature methods are shown in Table 8. Also, the stability analysis using the proposed model regarding the network metrics is given in Table 9. The table provides a comparison

Table 9 Routing stability analysis with different models

Methods	Delay (ms)	Throughput (Mbps)	Packet delivery rate (%)	Packet drop rate (%)	Network lifetime (s)	Jitter (ms)
ERS [16]	15	20	98	2	1000	5
DRL [17]	20	18	95	5	900	7
AODV [18]	30	15	90	10	800	10
ML-AODV [19]	25	17	93	7	950	8
ANN-SVM [20]	18	22	97	3	1100	6
RL [21]	22	19	94	6	920	7
DRL [22]	17	23	98	2	1200	5
Proposed	2.48	25	99	1	1300	4

Table 10 Comparison with baseline models

Methods	Delay (ms)	Throughput (Mbps)	Packet delivery rate (%)	Packet drop rate (%)	Network lifetime (s)	Jitter (ms)	Prediction accuracy (%)
LR	12.5	15	85	15	950	12	87.5
KNN	10	18	87	13	1000	10	91
FCNN	8.7	20	90	10	1050	8.5	93.5
CNN	5.3	22	94	6	1100	7	95.8
RNN	4.9	23	96	5	1150	6.2	97.2
Proposed	2.48	25	99	1	1300	4	99.25%

of various methods such as ERS, DRL, AODV, ML-AODV, ANN-SVM, RL, DRL, and Proposed in terms of their prediction performance metrics for a machine learning task in stable nodes prediction outcome. Compared to all the techniques, the proposed design attained higher accuracy and lower error percentages. The proposed method shows the lowest delay at 2.48 ms, significantly better than others, with AODV having the highest at 30 ms. The proposed method achieves the highest throughput at 25 Mbps. Other methods like DRL [22] and ANN-SVM [20] also perform well but are still lower than the proposed method. The recommended scheme has the highest delivery rate at 99%, with AODV lagging at 90%. The proposed method has the lowest drop rate at 1%, while AODV experiences the highest drop rate at 10%. The proposed method offers the longest network lifetime at 1300 s, which is a considerable improvement over AODV's 800 s.

The proposed method records the lowest jitter at 4 ms, indicating a more consistent performance compared to AODV's 10 ms. Overall, the proposed method outperforms the others across all key metrics, showing significant advancements in reducing delay, increasing throughput, improving packet delivery and drop rates, extending network lifetime, and minimizing jitter.

The designed model is also related to deep learning-based baseline models such as linear regression (LR), k-nearest neighbor (KNN), fully connected neural network (FCNN), CNN, and RNN. The comparisons are detailed in Table 10. The proposed model gained efficient results compared to the machine learning-based line models.

The baseline models were evaluated using a t-test and the statistical results were provided in Table 11. The t-statistic measures the difference in means between the designed method and each of the other methods. An increased t-statistic scores indicates a larger difference.

By analyzing historical data on node positions and movements over time, the DSR-CNN can learn patterns in how nodes typically move. This includes speed, direction, and frequency of movements, allowing the model to anticipate future positions based on past behaviour.

12.2 Discussion

The proposed solution outperforms prevailing mechanisms for routing in MANETs due to several key factors that address the limitations of conventional approaches. The designed DSR-CNN handles both spatial and temporal dependencies in MANETs. Traditional methods often rely on static or shallow learning models that fail to depict the dynamic character of the network, leading to suboptimal routing decisions. Enhanced Fick's Law Optimization improves the performance of the DSC-RNN by fine-tuning the network's parameters. This ensures that the neural network is not just trained but also optimized to minimize prediction error. GTO finds the most stable route by optimizing the selection of nodes with high stability scores. It reduces packet loss and latency by ensuring that routes are composed of reliable nodes. Unlike traditional routing algorithms that break when links fail and require re-routing, GTO continuously monitors and adjusts routes based on node stability. This minimizes disruptions and packet losses, providing better route recovery when links break.

However, the implementation of deep learning models is challenging for the resource-constrained MANET application. It requires extensive memories for parameters, intermediate states, and data that are not available on mobile devices. However, the leveraged Fick's law optimization optimized the parameters reducing the computational resource requirement during the training phase. Also, the model produces the solution at a faster convergence rate using the optimization procedures resulting in fewer iteration numbers for the model updates reducing the excessive resource usage. Furthermore, the model ensures the adaption to the network changes with minimal parameter variation. GTO facilitates efficient decision-making in selecting the optimal routing path by reducing the number of route calculations and re-calculations. This leads to fewer instances of high-computation routing processes, which saves both time and energy on mobile devices.

In this real-time scenario, the DSR-CNN model effectively maintains communication despite the dynamic and unpredictable nature of the network, providing stable, reliable routes that adapt to changes in real time. Imagine a rescue operation after a natural disaster, where a MANET is deployed for communication among rescue teams. The nodes are constantly moving, and the network topology changes as teams spread out. The proposed DSR-CNN model in this scenario would predict stable routes by analyzing the mobility and energy levels of the rescue team's devices, ensuring that communication is maintained through the most reliable paths, adapting to route failures when a team moves out of range, by quickly finding an alternative route through nearby teams, preventing communication breakdowns, optimize routing using the Giant Trevally Optimizer to reduce delays and packet loss, which is critical in delivering timely updates and commands to the rescue teams.

13 Conclusion

This study presents the DSR-CNN approach, a novel method for analyzing and optimizing stable end-to-end routes in MANETs. By leveraging Deep Supervised Learning and a Dynamic Sparse R-CNN, the proposed model effectively identifies stable nodes in dynamic environments, enhancing the reliability and connectivity of routing decisions. The incorporation of the Enhanced Fick's Law Optimization Algorithm and the Giant Trevally Optimizer allows for improved route exploration and recovery, significantly reducing packet loss and boosting overall network performance. The proposed model implementation demonstrates that the DSR-CNN model achieves the lowest average delay of 2.48 s with a standard deviation of 2.17, outperforming existing routing methods in terms of route stability and packet delivery efficiency. The utilization of the proposed research in the MANET communication shows 98% prediction accuracy, 2.48 ms delay, 25Mbps throughput, and 99% packet delivery ratio. Building on these promising results, future work will focus on integrating advanced machine learning models, such as graph neural networks, to further enhance the accuracy of node predictions. We also aim to optimize the route recovery mechanism by incorporating real-time data analysis and addressing energy efficiency in routing decisions. These enhancements are expected to improve MANET performance across various dynamic scenarios, paving the way for more resilient and efficient communication in mobile networks.

Authors contributions Gunavathie M A, Ujwal Ramesh Shirode—Conceptualization, Methodology, Software, Formal analysis, Investigation, Resources, Validation, Data Curation, Writing—Original Draft, Writing—Review & Editing, Supervision, Project administration. Nichenametla Rajesh, V. Sudha—Conceptualization, Methodology, Software, Formal analysis, Writing—Original Draft, Writing—Review & Editing, Visualization.

Funding No funding is provided for the preparation of manuscript.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

Consent for publication Every author who contributed to this work has given their full agreement for it to be published.

Ethical approval None of the authors of this article have conducted any research involving humans or animals.

Informed consent Every author has consented to contribute to this article.

References

1. Mani G, Prabakaran R. (2024) Artificial Neural Network Machine Learning Model for Malicious Action Detection and Prevention of MANET Assets. In2024 1st international conference on cognitive, green and ubiquitous computing (IC-CGU) pp. 1–6. IEEE.
2. Shrestha A, Sharma V, Hussein L, Aishwarya M, Satyanarayana AN, Saimanohar T. (2024) User Mobility Prediction in 5G Networks Using Recurrent Neural Networks. In2024 IEEE International conference on communication, computing and signal processing (IICCCS) pp. 1–6. IEEE.
3. Baumgartner M, Papaj J, Kurkina N, Dobos L, Cizmar A (2024) Resilient enhancements of routing protocols in MANET. Peer-to-Peer Netw Appl. 17(5):1–22
4. Hande JY, Sadiwala R (2024) Data security-based routing in MANETs using key management mechanism. SN Computer Sci 5(1):155
5. Rajalakshmi B, Anandhi RJ, Moorthi K, Kavin BP, Dhanaraj RK. (2023) Detection of Wormhole Attacks Using the DCNNBiLSTM Model to Secure the MANET. InInternational conference on computing and communication networks pp. 15–30. Singapore: Springer Nature Singapore.
6. Sarangi SK, Nanda A, Lenka R, Behera PK (2024) Hybrid heuristic driving training-rider optimization algorithm for QoS-aware multicast communication system in MANET. Aust J Electr Electron Eng 21(1):59–78
7. Nahar A, Das D (2023) MetaLearn: optimizing routing heuristics with a hybrid meta-learning approach in vehicular ad-hoc networks. Ad Hoc Netw 138:102996
8. Soundararajan S, Prabha R, Baskar M, Nagalakshmi TJ (2023) Region centric GL feature approximation based secure routing for improved QoS in MANET. Intell Autom Soft Computing. 36(1):267
9. Arif M, Sharm L, Jadhav VD, Revathi KG, Jyothi AP, Srivastava P (2024) Improving routing performance in mobile Ad Hoc networks using artificial neural networks for mobility prediction using deep learning. J Elect Syst 20(2s):942–949
10. Arun M, Jayanthi R (2023) An adaptive congestion and energy aware multipath routing scheme for mobile ad-hoc networks through stable link prediction. Measur Sens. 30:100926
11. Bhargava M, Chaturvedi A. (2024) Energy and Signal Strength based Reliable Route Prediction by Fuzzy Rules in MANET (ESRP). In 2024 fourth international conference on advances in electrical, computing, communication and sustainable technologies (ICAECT) pp. 1–7. IEEE.
12. Amalia A, Pramitarini Y, Perdana RH, Shim K, An B (2025) A Cross-Layer FL-based clustering protocol to support multicast routing in IoT-enabled MANETs with CF-mMIMO. IEEE Access. 13:3881–3899
13. Priyadarshi R (2024) Energy-efficient routing in wireless sensor networks: a meta-heuristic and artificial intelligence-based approach: a comprehensive review. Arch Computational Methods Eng 31(4):2109–2137
14. Koushik CP, Vetrivelan P, Chang E (2024) Markov chain-based mobility prediction and relay-node selection for QoS provisioned routing in opportunistic wireless network. IETE J Res 70(3):2233–2245
15. Murugan VS, Unhelkar B (2024) Optimizing mobile Ad Hoc network cluster based routing: energy prediction via improved deep learning technique. Int J Commun Syst 37(10):e5777
16. Zafar MH, Altalbe A (2021) Prediction of scenarios for routing in MANETs based on expanding ring search and random early detection parameters using machine learning techniques. IEEE Access 9:47033–47047
17. Abbood ZA, Atilla DÇ, Aydin Ç (2023) Enhancement of the performance of MANET using machine learning approach based on SDNs. Optik 272:170268
18. Duong TV (2024) An improved method of AODV routing protocol using reinforcement learning for ensuring QoS in 5G-based mobile ad-hoc networks. ICT Express 10(1):97–103

19. Baird I, Wadhaj I, Ghaleb B, Thomson C (2024) Impact analysis of security attacks on mobile Ad Hoc networks (MANETs). Electronics 13(16):3314
20. Shafi S, Mounika S, Velliangiri SJ (2023) Machine learning and trust based AODV routing protocol to mitigate flooding and blackhole attacks in MANET. Proced Computer Sci. 218:2309–2318
21. Militani DR, de Moraes HP, Rosa RL, Wuttisittikulkij L, Ramírez MA, Rodríguez DZ (2021) Enhanced routing algorithm based on reinforcement machine learning—a case of VoIP service. Sensors 21(2):504
22. Jothi Lakshmi S, Karishma M (2023) A modified DSR protocol using deep reinforced learning for MANETS. IETE J Res 70(5):1–2
23. Wheeb AH, Naser MT (2021) Simulation based comparison of routing protocols in wireless multihop adhoc networks. Int J Elect Computer Eng 11(4):3186
24. Wheeb AH, Al-Jamali NA (2022) Performance analysis of OLSR protocol in mobile ad hoc networks. iJIM. 16(01):107
25. Wheeb AH, Kanellopoulos DN (2020) Simulated performance of SCTP and TFRC over MANETs: the impact of traffic load and nodes mobility. Int J Bus Data Commun Netw (IJBDCN) 16(2):69–83
26. Mekkaoui A, Smail O, Rebbah M. (2024) Energy aware and link-Stability Routing Protocol based on Differential Evolution for MANETs. In 2024 international conference on networking and network applications (NaNA) (pp. 66–71). IEEE.
27. Pandey P, Singh R (2022) Efficient ad hoc on demand distance vector routing protocol based on route stability in MANETs. Int J Wirel Inf Netw 29(3):393–404
28. Gnanasekaran P, Varunkumar KA, Rajendran N, Priyadarshini R, Macherla S (2024) Optoelectronic-aided machine learning-based stable routing protocol for MANET and beyond massive MIMO systems in 5G networks. Opt Quant Electron 56(3):480
29. Ul Hassan M, Al-Awady AA, Ali A, Sifatullah AM, Iqbal MM, Khan J, Abdelrahman Ali YA (2024) ANN-based intelligent secure routing protocol in vehicular ad hoc networks (VANETs) using enhanced AODV. Sensors 24(3):818
30. Hong Q, Liu F, Li D, Liu J, Tian L, Shan Y. (2022) Dynamic sparse r-cnn. In proceedings of the IEEE/CVF conference on computer vision and pattern recognition pp. 4723–4732
31. Ebadinezhad S, Dereboylu Z, Ever E (2019) Clustering-based modified ant colony optimizer for internet of vehicles (CACOIOV). Sustainability 11(9):2624
32. Hashim FA, Mostafa RR, Hussien AG, Mirjalili S, Sallam KM (2023) Fick's law algorithm: a physical law-based algorithm for numerical optimization. Knowl-Based Syst 260:110146
33. Sadeeq HT, Abdulazeez AM (2022) Giant trevally optimizer (GTO): a novel metaheuristic algorithm for global optimization and challenging engineering problems. Ieee Access 10:121615–121640