Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.064102

ARTICLE

Distributed Computing-Based Optimal Route Finding Algorithm for Trusted Devices in the Internet of Things

Amal Al-Rasheed[1], Rahim Khan[2,*], Fahad Alturise[3] and Salem Alkhalaf[4]

[1]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[2]Department of Computer Science, Abdul Wali Khan University, Mardan, 23200, Pakistan
[3]Department of Cybersecurity, College of Computer, Qassim University, Buraydah, 52571, Saudi Arabia
[4]Department of Computer Engineering, College of Computer, Qassim University, Buraydah, 52571, Saudi Arabia

*Corresponding Author: Rahim Khan. Email: rahimkhan@awkum.edu.pk

Received: 05 February 2025; Accepted: 28 March 2025; Published: 09 June 2025

ABSTRACT: The Internet of Things (IoT) is a smart infrastructure where devices share captured data with the respective server or edge modules. However, secure and reliable communication is among the challenging tasks in these networks, as shared channels are used to transmit packets. In this paper, a decision tree is integrated with other metrics to form a secure distributed communication strategy for IoT. Initially, every device works collaboratively to form a distributed network. In this model, if a device is deployed outside the coverage area of the nearest server, it communicates indirectly through the neighboring devices. For this purpose, every device collects data from the respective neighboring devices, such as hop count, average packet transmission delay, criticality factor, link reliability, and RSSI value, etc. These parameters are used to find an optimal route from the source to the destination. Secondly, the proposed approach has enabled devices to learn from the environment and adjust the optimal route-finding formula accordingly. Moreover, these devices and server modules must ensure that every packet is transmitted securely, which is possible only if it is encrypted with an encryption algorithm. For this purpose, a decision tree-enabled device-to-server authentication algorithm is presented where every device and server must take part in the offline phase. Simulation results have verified that the proposed distributed communication approach has the potential to ensure the integrity and confidentiality of data during transmission. Moreover, the proposed approach has outperformed the existing approaches in terms of communication cost, processing overhead, end-to-end delay, packet loss ratio, and throughput. Finally, the proposed approach is adoptable in different networking infrastructures.

KEYWORDS: Internet of things; distributed communication; security; authentication; decision tree

1 Introduction

The Internet of Things (IoT) refers to the smart infrastructures where devices with embedded sensors collaborate to form a network. IoT facilitates reliable communication and data sharing among distributed modules, i.e., devices and servers. For this purpose, a secure communication methodology is used to ensure the timely transmission of packets over a shared and insecure communication medium [1]. For this purpose, two general methodologies, i.e., direct or hop-based transmissions, were reported in the literature. The selection criteria of these approaches have a direct proportionality ratio to the application requirements of IoTs. The direct communication of member devices with the intended server module is possible only if deployed within the server module's coverage area [2]. This scenario is possible only if the traffic strategy

is engineered and the registration process of vehicles is applicable [3]. However, engineering deployment is not always realistic [4,5]. Therefore, multi-hop-based communication methodologies are adopted to ensure reliable packet transmission from the source to the respective server module via a relay device.

In multi-hop communication, every device, i.e., the source module, finds an optimal path before initiating the actual transmission process. For this purpose, different methodologies have been reported in the literature. Initially, the shortest path-based communication methodologies were reported, and every device had to transmit packets via that path in general and IoT in particular. The idea is to mimic the behaviors of human beings adopted in traffic [6]. However, a common issue with the shortest-path-enabled approach is the rapid consumption of the available power by the device deployed on it, as it has the highest probability that multiple devices will use this path for communication, just as the shortest path has more traffic than longer paths. To address this issue, optimal path-enabled methodologies have been reported where sensors or devices forward captured data via reliable paths. However, these techniques have the same problems as multiple devices or sensors, where devices will likely share the same optimal path [7]. Therefore, multi-path-based communication approaches were developed to address the common path issue. In these methodologies, every device or sensor holds valuable information about neighboring modules, such as residual energy, hop count, criticality factor, and number of dependent modules in IoT. These methodologies have resolved the core issue, which is linked with both shortest and optimal-path-based approaches. In these approaches, the traffic generated by the source modules is uniformly distributed across multiple available paths in IoT. An optimized multiple-path oriented routing and node deployment methodology has been introduced to enhance the availability of the underlined network [8]. However, this model is limited to smart city application areas. A decision-tree-enabled distributed traffic management system is presented to ensure the challenging problem of route flopping [9]. However, packet pin-pong is a common issue with this scheme as the hop-count value of the devices is not utilized in the classification. Likewise, a random forest and particle swarm optimization-based classification scheme was presented by Lavate et al. [10]. Likewise, a hybrid classification approach is based on Cuckoo Search and PSO along with three classifiers: Multi-Layer Perceptron, AdaBoost, and Random Forest [11]. However, this model doesn't fit the resource-constrained IoT devices due to its highest complexity. A convolution neural networks-based intrusion detection model was presented to resolve various issues associated with existing federated learning-enabled approaches, i.e., transmission of model parameters and leakage of private data. However, hard-labeled strategies and voting mechanisms are questionable for secure IoT [12]. Additionally, existing communication methodologies have limitations such as the longest transmission delay and packet loss ratio, but are also vulnerable to numerous intruder attacks in an active IoT networking infrastructure.

In this paper, a reliable multi-hop-enabled communication mechanism is developed to resolve the aforementioned issues with the existing algorithms or methodologies, especially those designed for IoT. The proposed methodology is based on the realistic assumption that every device should be well-informed about neighboring devices, especially those operating in its coverage area. To do so, devices gather valuable information about neighboring devices' distance, criticality factors, link reliability, neighboring devices, delay factors, etc. Every source module uses these metrics to find an optimal path toward the respective server or relay device. The main contributions of the paper are given below:

1. A machine learning-enabled device-to-device and device-to-server authentication algorithm.
2. A reliable communication methodology in which the source and destination devices or servers shouldn't be operated in the direct coverage domain of the respective transceiver's module.
3. Packets are transmitted in encrypted form to ensure data confidentiality and privacy in the active IoT.
4. A trustworthy communication infrastructure that is particularly designed for the IoT.

The remaining paper is arranged as follows.

In the subsequent section, i.e., the literature review, a summary of the existing communication methodologies, specifically those developed for the IoT, is presented. In the next section, a detailed description of the proposed methodology is presented, such as how devices are enabled to be informed about their surroundings and find an optimal path. Verification of various claims is carried out through extensive simulations, which are presented in the performance evaluation section of the paper. Finally, the concluding remarks are given.

2 Literature Review

In the literature, various approaches have been developed to ensure reliable communication among active devices in IoT. A detailed summary of the most relevant approaches is presented here.

Collaborative communication approaches were designed and implemented to ensure reliable transmission of data among the integrated and sensing-enabled devices. In this approach, a relay device serves as a mediator between another device and the server module if and only if the late device doesn't have direct communication capacity with the nearest server. These devices not only transmit their data but also forward packets to the neighboring devices. A hybrid communication module that is based on the trust model and multiple path packet forwarding was presented to develop trusted communication sessions between interested devices [13]. Likewise, a task offloading methodology, preferably in a distributed manner, was presented for computationally expensive tasks to ensure a balanced trade-off between processing and consumed energy overheads [14]. Moreover, deep learning-based methodologies were developed to ensure a balance between processing and energy cost [15]. For this purpose, authors have assumed that the IoT infrastructure had a random flow of traffic and a dynamic networking environment. However, the compatibility of deep learning-based algorithms is questionable. In addition, a sophisticated communication and task-offloading methodology was presented in [16] to make sure that optimization of processing resources, channel allocation for reliable communication, and task scheduling was carried out with the minimum possible value of delay metrics. Kalman filter-oriented algorithm was presented in [17] to find the approximate position of the device. This approach had the flexibility to be adopted in different applications with minimum possible tracking beam overhead, a common issue linked with the earlier approaches. This approach is safe against various intruder attacks as it has to rely on the existing security models. Traffic or route awareness plays a significant role in the IoT. Therefore, communication approaches with embedded authentication and privacy were reported [18,19]. This scheme has enabled a service-aware communication infrastructure for the devices along with additional flexibility, that is, an on-demand task scheduling policy. Similarly, the genetic algorithm-based algorithm was presented to resolve premature and convergence issues [20]. Likewise, collaborative fusion-enabled methodologies were presented to reduce the overall communication overhead and congestion across different available paths in the IoT, but security and device trustworthiness are among the challenges linked with it. An ISCC-enabled fusion methodology was presented in [21], where every device must send data in a refined form, i.e., aggregated with the minimum possible duplicate and missing values. A secure and lightweight packet transmission methodology was introduced by Ibraheem et al. [22], where a symmatric key and elliptic curve were combined to form a hybrid authentication model for IoT. However, complexity is among the core challenges with this approach.

A federated learning and Green computing-enabled optimal route selection scheme was introduced by Khatua et al. [23], where a genetic algorithm was integrated with the route selection process to find the optimal one. Optimal routes are computed by the concerned server and local server module and are shared with the device, preferably within the defined time bond. However, the implementation of a two-tier approach for servers and the sharing of information are the challenges associated with it. Likewise, a

server and reinforcement-learning-aware route prediction mechanism was developed to ensure that every device should have sufficient knowledge about the operating area [24]. Likewise, a block-chain and AODV based security schemes have been presented [25,26]. Similarly, references [27,28] have introduced secure approaches for 6G and internet of Vehicles. These models have a high susceptibility ratio to attacks launched by the potential intruder modules. However, a common issue with the existing approaches is that optimal routes are computed either by server or server module, which is not realistic in the dynamic IoT environment. Thus, a reliable optimal route optimization scheme should be developed where route computation is carried out by the concerned device.

3 Proposed Secured Communication Approach for the Internet of Things

The proposed secured communication model is designed for the Internet of Things, where devices and servers are trained through a sophisticated AI-enabled process. The proposed scheme ensures reliable communication between interested parties, i.e., devices and server modules, and is safe against intruder attacks in the IoT. The proposed communication approach is adaptable to both networking infrastructures, that is, (i) homogeneous, where multiple-hop communication strategies are used, and (ii) heterogeneous, where both direct and multi-hop communications are applicable. Apart from reliable communication, the proposed scheme has been secured through a lightweight authentication approach, i.e., device to server. A detailed description of these techniques is given below.

3.1 Proposed Communication Approach for IoT

Generally, in resource-constrained networking infrastructures such as the Internet of Things and Artificial Intelligence-enabled Internet of Things, embedded devices operate on roads to capture real-time information about the underlined phenomenon, process and share it with the centralized module, i.e., server module, either directly or through the neighboring devices. However, in scenarios where these devices are allowed to join randomly, ensuring that every device is running within the coverage area of the particular server module is tough. Therefore, a multi-hop communication methodology is used where packets are forwarded through the optimal neighboring devices, which are based on various parameters, i.e., residual energy E_r, Receive Signal Strength Indicators (RSSI), Success ratio, average transmission time, and load. Moreover, neighboring devices are bound to share these statistics after a defined interval of time.

Initially, the respective server module generates a simplified message with an embedded data field of hop and broadcasts it. It is received by those devices operating in the coverage area of the server module's transceiver using the Eqs. (1) and (2), respectively.

$$device_{list} = Function(Device_{ID}) \tag{1}$$

Function is used to capture IDs of all neighboring devices, especially those deployed in the coverage area of the transceiver.

$$Bcast = Send(R_{ID}, (device_{list}, msg)) \tag{2}$$

R_{ID} represents the identity of the source module, which is the server in this case. As soon as a neighboring device receives this message, it updates the message contents, that is hop count value is set to one (1), and prepares to rebroadcast it. However, multiple neighboring devices may broadcast updated versions

of the message, which leads to the packet collision, for which CSMA/CA is used as shown in the Eq. (3).

$$Efficiency(\eta) = \frac{T_t}{e * 2 * T_p + T_t + T_p} \tag{3}$$

T_t & T_p represent transmission and propagation time, respectively. This process is repeated by every device (i.e., from the device with a hop count of one to devices with values of two, three, four, and so on) until the very last device has computed its hop count value. Hop count values play a crucial role in defining the neighborhood or load of a device. This is a time-consuming process, but it is acceptable as it requires to be carried out only once, preferably after the deployment.

If the neighborhood discovery process is successful, then devices are bound to share valuable information about other crucial parameters such as success ratio, transmission time T_p, and residual energy. Additionally, RSSI values of the neighboring devices are computed by the respective device itself as presented in the Eq. (4).

$$RSSI = P_t - Path_{Loss}(d) \tag{4}$$

P_t and d represent the devices' average propagation time and overall distance. Secondly, $Path_{Loss}$ is the path loss, as shown in the Eq. (5).

$$Path_{Loss}(d) = Path_{Loss}(d_0) + 10nlog\left(\frac{d}{d_0}\right) + X_\sigma \tag{5}$$

d_0 represents the reference distance and X_σ the dB. dB is the unit used to measure the signal strength loss ratio.

Once every device V_i has information about neighboring devices, then it is ready to be trained on the respective machine learning model, i.e., decision tree. The decision tree is a supervised machine learning technique where data is required to be labeled, that is, in this case, such as known parameters and output, i.e., optimized or not. Moreover, a decision tree is an ideal solution for scenarios where decisions are based on various parameters in sequential order. Secondly, optimal devices should be identified from the neighboring devices, and a threshold value of δ is needed. Thus, if the computed value of the expression is less than or equal to the threshold value, then the neighboring device is considered optimal or non-optimal. For this purpose, weights are assigned to every parameter, and the optimal ratio of the neighborhood is checked through a sophisticated process. In the proposed setup, RSSI value and neighborhood or load are assigned the weights of ω_1 and ω_1, respectively.

Initially, neighboring devices were classified into two classes, i.e., (i) potential relaying devices, which could be used as an intermediate device for the transmission of packets, and (ii) neighbor only, a device that has a higher hop count value than the source device. In the decision tree, entropy is the information necessary to define the similarity of the data values in a data set, as described in the Eq. (6) given below.

$$Entropy = -\sum_{i=1}^{n} p_i * log(p) \tag{6}$$

The data set used in the proposed set has a relatively higher entropy value, i.e., those values are approximately equally divided. In addition to entropy, the Gini Index is used to describe the homogeneity of

data values, and a parameter with a slightly higher value could be used as the root node or basic classifier in the decision tree. The Gini index of parameters is computed using the Eq. (7).

$$Giniindex = 1 - \sum_{i=1}^{n} p(i)^2 \tag{7}$$

where n represents classes, i.e., optimized or not, and $p(i)$ is the approximate optimal neighbors ratio to the total neighbors.

In the proposed setup, RSSI has a slightly higher value of the Gini Index along with allocated weightage, i.e., 30%, and is used as a root node in the decision tree where neighboring devices are classified as potential routers, i.e., can forward packets of neighboring devices in addition to its own, or not. For this purpose, a threshold value of the RSSI value is defined, and only those devices with maximum values are among the potential candidates for packet forwarding, subject to other parameters in the IoT. Secondly, RSSI serves as the root node of the decision tree. Secondly, hop count has a higher value in the Gini Index than other parameters except RSSI and is used in the decision tree where neighboring devices are classified as potential routers, i.e., can forward packets of neighboring devices or not. For this purpose, hop count values of the neighboring devices are fed to the internal node, i.e., level-1, in the decision tree. A device with a hop count value less than or equal to the hop count value of the source is then added to the class where potential routing devices are stored. Otherwise, it is classified as an ordinary neighboring device and avoids being used as a relay device.

The third parameter, which has a greater value than the remaining parameters, is the residual energy, i.e., E_r, of a neighboring device, which is defined as available power for the smooth processing of the device. Thus, the class of potential relaying devices is further refined by the residual energy parameter, and neighbors with maximum residual power are allowed. For this purpose, a threshold value has been defined, i.e., 50%, in this case. However, in scenarios where all devices in the list have residual power less than 50%, then another threshold value, i.e., 90%, is utilized. Finally, the next level node in the decision tree is based on two remaining parameters, i.e., average transmission time and load. However, as these parameters have a slightly different range of values, data should be normalized before actual processing. For effective normalization of data, the following Eq. (8) has been utilized.

$$Normalized(data) = \frac{X - min}{max} \tag{8}$$

where current, minimum, and maximum values are represented by X, min, and max, respectively. After this, normalized values are fed into the Eq. (9) given below:

$$optimal(V_{ngb}) = \omega_1 * T_p + \omega_2 * Load \tag{9}$$

where V_{ngb} represents neighboring devices of the source. ω_1 & ω_2 represent weights assigned to these parameters based on the importance of routing methodology. A neighboring device is optimal if it has the minimum possible value obtained through the Eq. (9) among all neighbors.

An algorithm for the proposed machine learning-enabled communication infrastructure has been presented below. The complexity of the proposed algorithm is **O(2n+m)**, where n and m represent the complete set of devices and servers, respectively.

Algorithm 1: Finding an optimal neighboring device in artificial Intelligence-enabled internet of things

Require: Optimal Neighboring device.
Ensure: Trusted and Optimized
1: C_i complete set of devices in IoT
2: S_j complete set of server in IoT
3: E_r residual power of devices
4: H_c hop count value of devices
5: $T_p(C_{opt}) = 1200\ u\ sec$ Propagation time of devices
6: $Load_N = 100$ Dependent Neighboring devices
7: *RSSIi* Approximate value of the RSSI
8: *ClassPN* Potential Neighbors
9: *RSSIi* Approximate value of the RSSI
10: **for** (i = 0; i >= n; i++)
11: **if** RSSI(C_i) >= Threshold **then**
12: Add $Class_{PN} \leftarrow C_i$
13: **else then**
14: Skip C_i
15: **endif**
16: **endfor**
17: **while** ($C_i \in Class_{PN}$)
18: **if** $H_c(C_i) <= C_{MP}$ & $E_r >= \delta$ **then**
19: move to Next Neighbor
20: **else then**
21: Remove C_i from $Class_{PN}$
22: **endif**
23: **if** ($T_p(C_i) <= T_p(C_{opt})$) & Load(C_i) <= $Load_N$ **then**
24: $T_p(C_{opt}) \leftarrow (T_p(C_i)$
25: $Load_N \leftarrow$ Load(C_i)
26: $Optimal_{Nbr} \leftarrow C_i$
27: **else then**
28: Remove C_i from $Class_{PN}$
29: **endif**
30: **endwhile**
31: **if** (Multiple($C_i \in$) $Optimal_{Nbr}$) **then**
32: $Optimal(C_i) = Random(Optimal(C_{1,2,..,n}))$
33: **endif**
34: **return** Optimal Neighboring device C_i

Initially, devices are passed through the training phase, where data values related to both scenarios, optimal and non-optimal, are present, i.e., the IoT traffic data set [29], which has 10,000 records for various smart devices. For this purpose, the data set was divided into training and testing data values. (i) The training data set consists of 5000 records with the aforementioned parameter values. (ii) The testing data set consists of 5000 records. Moreover, a 5-fold cross-validation mechanism makes the result consistent and realistic. A random search-enabled hyperparameter-tuning methodology was used where required. It is important to note that missing values were also present in the data set, which were refined before the proposed solution's

training phase. To realize this, a simplified refinement scheme, preferably a well-known state-of-the-art existing approach with a high accuracy and precision ratio, has been utilized. As soon as the data set is refined, the training process of the devices with the proposed decision-tree-enabled algorithm, given above Algorithm 1, is carried out. Every device is trained to ensure that it can handle both scenarios, i.e., the one where an optimal neighbor is identified and situations where two or more optimal neighbors are identified. Two or more neighboring devices may have similar optimal values, i.e., for example, 0.5, then the question is which of these is selected as a relaying device. A randomized selection procedure is used to handle this situation, as shown in the Eq. (10).

$$Optimal(C_i) = Random(Optimal(C_{1,2,\ldots,n})) \tag{10}$$

where $Optimal(C_{1,2,\ldots,n})$ represents neighboring devices with similar optimal values, those computed through the proposed decision-tree-enabled communication approach.

After the successful completion of the training phase, the proposed model was thoroughly tested through a tested data set, i.e., both from benchmark and real-time that were generated during the simulation setup of the proposed scheme. During the testing phase, we investigated various scenarios, such as (i) if a device falls within the coverage area, then it shouldn't execute the proposed algorithm as it doesn't need to find an optimal neighbor, (ii) devices with multiple optimal neighbors, (iii) devices with two or more devices with the same RSSI values, etc. The proposed decision tree-enabled communication approach has performed exceptionally well in almost every possible scenario that was carried out during the test phase.

3.2 Proposed Light-Weight Authentication Scheme for the Decision Tree-Enabled Internet of Things

Generally, communication activity, i.e., device-to-device or device-to-server, is carried out through a shared and non-secure transmission medium. Therefore, every packet is transmitted in the encrypted form where both source and destination modules have a unique secret key, i.e., λ_i if any. Secondly, the intruder has intercepted the encrypted message in the IoT. Usually, intruders have two different objectives, i.e.,

1. Read information, but it doesn't affect the original message contents.
2. Make use of the information and update the original message, i.e., affects the system.

Therefore, authentication schemes should be pruned against both attacks in the realistic environment of IoT. A strong encryption mechanism should be adopted to convert plain text into equivalent cipher text. However, the authenticity of the source module should be checked before initiating actual communication sessions. Therefore, the authenticity of both parties, i.e., source and destination module, is carried out before transmission of the actual data, which is captured by legitimate devices. Thus, if a device, i.e., source or destination, has the expected security level, i.e., the device on the other side is trusted, then it can trigger the communication process; otherwise, abort it. For this purpose, a simplified authentication scheme is presented in this section to ensure that only trusted devices can communicate.

In addition to other parameters, every device must share its MAC address with the neighboring devices. Every device stores the MAC address of the neighbors, which is used to ensure the authenticity of the requesting module. In addition to the MAC address, time stamp information is collected. That is the approximate time required for a packet to be received from the dependent neighboring device, any device that isn't within the coverage area of the respective server module. These two parameters are used to ensure the legitimacy of the requesting dependent device. Moreover, the learning model, which generates weights from the respective MAC addresses, is shared in the hop count discovery phase. Therefore, instead of appending the actual MAC address to the packet header, its corresponding weight, which is computed through a sophisticated machine-learning-based model, is appended. This mechanism is not only

involved in improving the security of the model, but it is equally important in separating legitimate and adversary modules.

Thus, if a packet is received from the dependent device, the relay device should check its legitimacy by looking at the weights appended to the packet. Every device has MAC addresses and weights of the respective dependent or neighboring devices. Therefore, this weight is compared with stored weights and MAC address. If a match is encountered, the requesting device is assumed to have cleared the initial security or authenticity check, as depicted in the Eq. (11).

$$L_1 = Weight(V_i) \in Stored(Weights \ \& MACAddresses) \tag{11}$$

L_1 represents the first level of authenticity of the source device. Where $MAC\ Class$ holds the addresses of the authentic dependent modules shared in the hop-count discovery phase. If the MAC address of the concerned device belongs to the stored addresses class, then it is partially assumed to be authentic as it has passed the first security barrier in the proposed setup. Secondly, the timestamp information of the requesting device received in the packet is matched with the approximate timestamp information stored in the memory of the destination module, as depicted in the Eq. (12).

$$L_2 = timestamp \ \in approximate\ timestamp\ class \tag{12}$$

L_2 represents the second level of authenticity of the source device. Although this mechanism is simple, it is safe against well-known intruder attacks on networking infrastructure. Secondly, this scheme should be embedded with a well-known encryption scheme, thus making it hard for the intruder module to deceive the destination module, i.e., devices and servers. The algorithm for the proposed authentication methodology is depicted in Algorithm 2, where V_i & R_j represent devices and servers. $Class(MAC_{Alias})$ stores valuable information about numerous aliases, especially those generated through the proposed decision-tree-enabled authentication algorithm. This algorithm, i.e., Algorithm 2, bounds every device and server to verify the trustworthiness of the respective module before triggering the communication activity in the working domain. The proposed algorithm verifies the authenticity of the requesting device through two different parameters, i.e., the MAC alias generated through machine-learning-based methodology where the second parameter is the approximate propagation time. The complexity of the proposed authentication algorithm is **O(n)**, where n represents a complete set of devices and servers in the underlined IoT.

Algorithm 2: Authentication process of the communicating devices in the internet of things

Require: Separation of Legitimate devices from the Intruders.
Ensure: Trusted device
1: C_i complete set of device in IoT
2: S_j complete set of servers in IoT
3: Class MAC_{alias} ← Alias and MAC of C_{nbr}
4: Tp Approximate Propagation Time of C_i
5: A_k Intruder Modules
6: **for** (i = 0; i >= n; i++)
7: **if** $(MAC_{alias}(C_i)) \in MAC_{alias}$ &$\Delta(T) \in Class_T$ **then**
8: Requesting device is Trusted
9: Send a Response Message
10: Allowed to Start Communication

(Continued)

Algorithm 2 (continued)

```
        else then
            device is an Intruder
            Block its ID and MAC
            Shares its Info with Neighbors
        endif
    endfor
return Trusted Module C_i
```

4 Results and Evaluations of the Proposed Decision Tree-Enabled Methodology

Generally, a newly developed methodology should be supported by a detailed and thorough evaluation, particularly through well-known performance metrics and comparisons with existing state-of-the-art approaches. To do this, experimental or simulation results, whichever is applicable and feasible, should be presented in the working domain of IoT. Therefore, the proposed decision-tree-enabled authentication and communication methodology is implemented in NS-3, an open-source software for resource-limited networks. Initially, an IoT network was developed where device and server modules were deployed per the general deployment strategies available in the literature. The transmission range, T_p, of every device transmitter is approximately 450 m in the presence of obstacles. A detailed description of other parameters is given in the following Table 1.

Table 1: Secure internet of things parameters setup

Parameter name	Approximate or exact value
Approximate Area of IoT	500 × 500
Simulation software	NS-3
Devices	40–350
Servers	5%
Device's Transmission interval	20 u Sec
Link bandwidth	10 Mbps
Transmission cost (Packet)	75.6 mW
Receiving cost (Packet)	75.6 mW
Idle state consumption	1.2 mW
Sleep mode consumption	0.7 μW
Residual energy	Available power
Coverage area of XBEE module	450 M
Packet Size	128 Kbps
Type of the network traffic	UDP and CBR
Transmission interval	30 s
Forwarding interval	Immediate
Topology	Random deployment
Deployment	Random and engineered

4.1 Average Power Consumption of Devices in the Proposed Secured Communication Approach

In networking infrastructures where resources are limited, i.e., the Internet of Things and Wireless Sensor Networks, etc., effective usage of the available power source is crucial to ensure that devices are active for the maximum duration. Secondly, efficient utilization of the available power source is required to keep the networking infrastructure active for the maximum possible duration. The device's power consumption is computed using the Eq. (13).

$$vg_P(V_i) = P_{Cost} + Comm_{Cost} \tag{13}$$

P_{Cost} represents the overall processing cost, i.e., from capturing data values to forwarding them to the XBee module, and $Comm_{Cost}$ refers to its transmission by the appropriate module. A detailed comparison of the proposed and existing state-of-the-art approaches, particularly concerning the average power consumption, is shown in Fig. 1. The graphical results verify the supremacy of the proposed secured communication approach by completing similar activities using the minimum possible energy. Additionally, error bars and statistical significance are shown in Fig. 2.

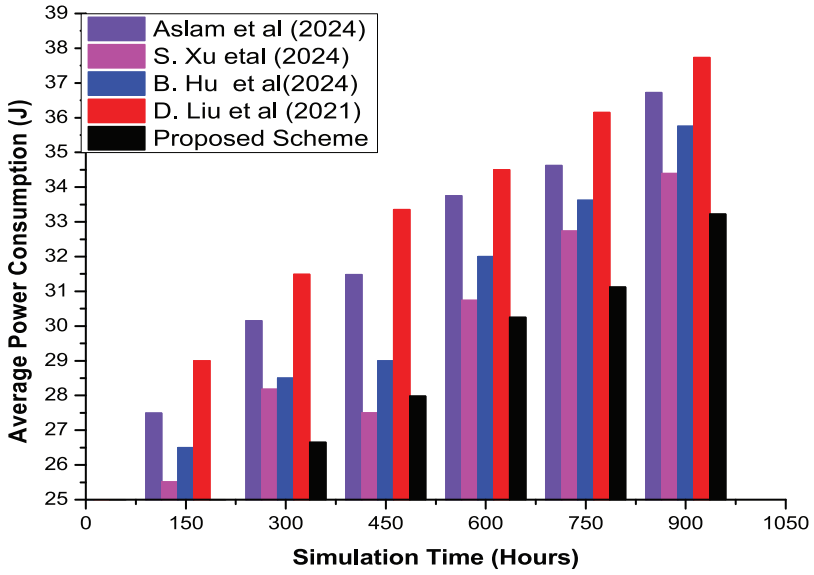

Figure 1: Average power consumption ratio of devices in the proposed secured communication infrastructure [4,5,7,9]

4.2 End-to-End Delay of the Proposed Secured Communication Approach

End-to-end delay is another important metric to measure the superiority of the newly proposed scheme, especially from the communication perspective, over existing approaches for the underlined networking infrastructure. This parameter directly correlates with the overall processing and packet transmission overheads; thus, a communication approach with minimal overhead is preferred. Therefore, a comparative analysis of the proposed and existing approaches is shown in Fig. 3, which indicates the exceptional performance of the former scheme over the latter one. This minimal cost overhead is possible due to the selection process, which is based on a proper assessment and requirement-oriented parameters incorporated into the hybrid formula. Secondly, the proposed scheme has ensured the minimum possible end-to-end delay is achieved while maintaining maximum security against well-known intruder attacks.

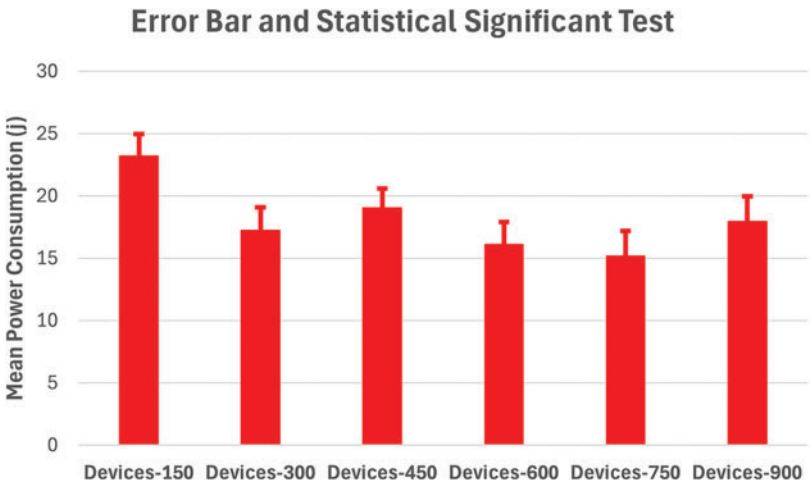

Figure 2: Statistical significance in terms of average power consumption

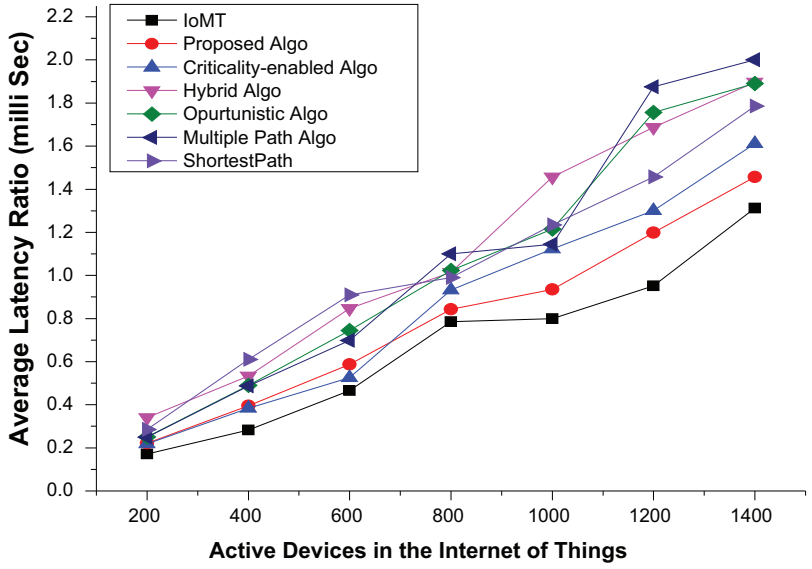

Figure 3: Average end-to-end delay ratio of devices in the proposed secured communication infrastructure for IoT

4.3 Average Packet Loss Ratio of the Proposed Secured Communication Approach

In addition to end-to-end delay metrics, the proposed and existing approaches are thoroughly investigated using other performance evaluation metrics, that is, the average ratio of successfully delivered packets. Generally, the packet loss ratio is increased due to the maximum likelihood of packet collision, channel interference, and deployment of devices in an area that is not accessible directly. The proposed approach has a higher APDR ratio than existing approaches, as shown in Fig. 4 by reducing the collision ratio through a time slice-oriented approach where every device should communicate only in the allocated time stamp. Secondly, devices are deployed where every device should communicate directly with the nearest server or through a neighboring relay device.

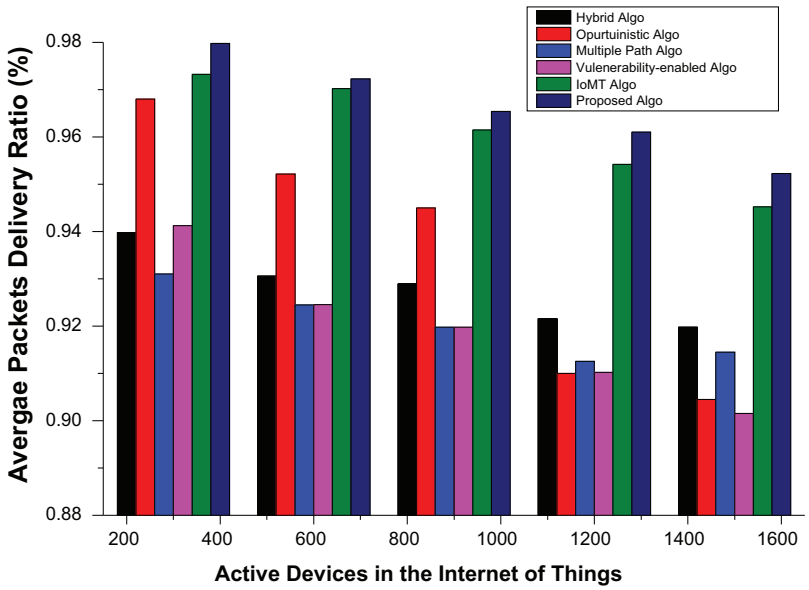

Figure 4: Average packet loss ratio of devices in the proposed secured communication infrastructure for IoT

4.4 Approximate Processing Cost of Devices in the Proposed Secured Communication Approach

Processing cost overhead is not only associated with the communication approaches but is also affected by a device-level authentication implementation or security scheme. This overhead is directly correlated to the number of operations required to be completed to verify the authenticity of the requesting module. Fig. 5 presents a detailed graphical comparison of the proposed and existing approaches, which shows the exceptional performance of the former approach over the latter ones in the realistic environment of IoT. The processing overhead of the proposed scheme is minimal as it relies on the minimal set of parameters required to verify the authenticity of the requesting module.

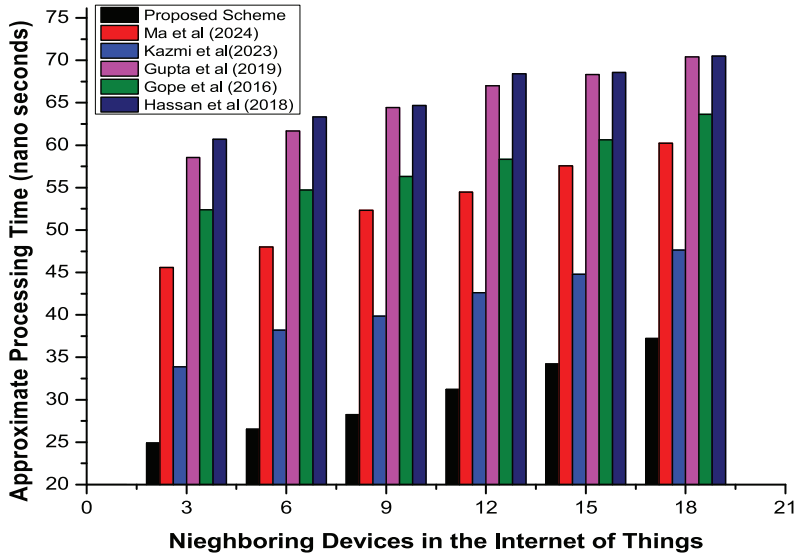

Figure 5: Average processing time overhead of devices in the proposed secured communication infrastructure for IoT [19,20,25–27]

4.5 Approximate Communication Cost of Devices in the Proposed Secured Communication Approach

Communication cost overhead is the number of bits transmitted to ensure mutual authentication of both devices and server modules. A detailed comparison of the proposed authentication and existing approaches is depicted in Fig. 6, which shows that the former scheme has outperformed the latter approaches, especially in communication overhead. Moreover, the proposed approach has achieved this without compromising the overall security of devices.

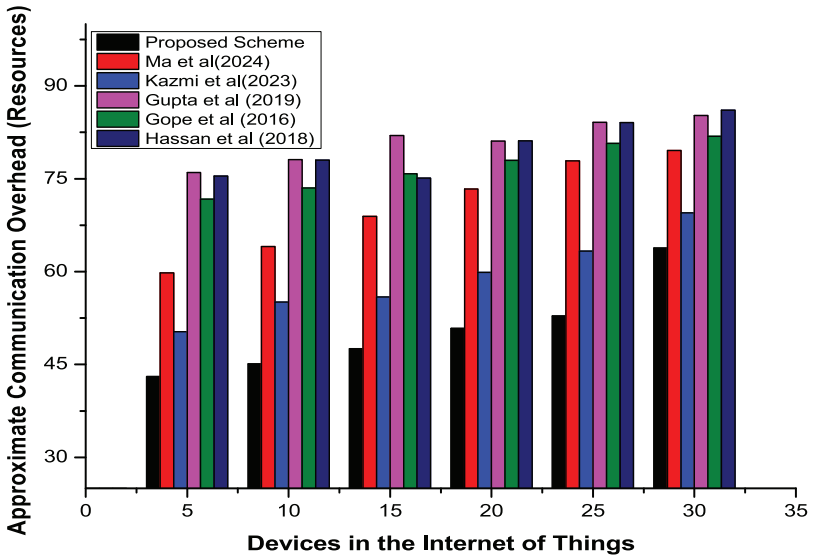

Figure 6: Average communication cost overhead ratio of devices in the proposed secured communication infrastructure for IoT [19,20,25–27]

4.6 Security Analysis

The proposed authentication scheme has tight security measures, which make it safe against various intruder attacks, device and server impersonation, reply, denial of services, etc.

1. The proposed algorithm guarantees the safety of the devices from an intruder module trying to deceive a legitimate module, i.e., device and server, by pretending to be a legitimate and trustworthy module. Every message passes through a rigorous and complicated encryption process, which makes it hard for the adversary to extract valuable information.
2. The proposed algorithm is safe against eavesdropping attacks as every packet is encrypted with a secret key.
3. Likewise, it is safe against perfect forward & backward secrecy attacks as intruder modules cannot convert the cipher text into plain text in a defined time interval.
4. Denial of Services (DoS) attacks are among the core challenges linked with the authentication schemes, especially those designed for the IoT. The proposed approach has the necessary measures against the DoS attack. The encryption algorithm is strong and beyond the operational capabilities of the resource constraint devices to decipher an intercepted message. Secondly, if an intruder module tries to bombard the server through a dummy packet, these are easily identified as every encrypted message has a unique pattern.

5 Conclusion

Technological advances in sensors and actuators led to a collaborative networking infrastructure of devices and servers where devices share collected data with the intended server module. In this network, devices coordinate with each other to establish a smart and collision-free traffic environment. For this purpose, every module, i.e., device or server, communicates via a reliable transmission channel and shares valuable information about the traffic conditions on the various available routes. However, direct communication with the nearest server module is not always possible. Therefore, an alternate communication strategy is adopted where certain devices, particularly those running in the servers' coverage area, mediate between the source device and the intended server. For this purpose, a neighborhood knowledge-based communication approach was presented in this paper that enables devices operating in the out-of-range area to share their information with the intended server module. In this approach, devices must share data such as distance, RSSI value, link reliability, dependent devices, etc. Moreover, a secure key-based security algorithm was integrated to ensure a reliable transmission of packets. Simulation results concluded that the proposed multi-path-based communication approach could be an ideal methodology for IoT where distance among servers is kept at a maximum level. The proposed scheme has achieved maximum throughput with the minimum possible average transmission delay in the IoT.

Acknowledgement: This research work is supported by the Princess Nourah bint Abdulrahman University Riyadh, Saudi Arabia, through Project number (PNURSP2025R235).

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2025R235), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design: Amal Al-Rasheed and Rahim Khan; data collection: Amal Al-Rasheed, Rahim Khan and Fahad Alturise; analysis and interpretation of results: Amal Al-Rasheed, Rahim Khan, Salem Alkhalaf; draft manuscript preparation: Amal Al-Rasheed, Rahim Khan, Salem Alkhalaf and Fahad Alturise. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: This article does not involve data availability.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Al-Ani AK, Ul Arfeen Laghari S, Manoharan H, Selvarajan S, Uddin M. Improved transportation model with internet of things using artificial intelligence algorithm. Comput Mater Contin. 2023;76(2):2261–79. doi:10.32604/cmc.2023.038534.
2. Dargaoui S, Azrour M, El Allaoui A, Guezzaz A, Alabdulatif A, Alnajim A. Internet of things authentication protocols: comparative study. Comput Mater Contin. 2024;79(1):65–91. doi:10.32604/cmc.2024.047625.
3. He Y, Ma L, Cui J, Yan Z, Xing G, Wang S, et al. Automatch: leveraging traffic camera to improve perception and localization of autonomous vehicles. In: Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems. Boston, USA; 2022. p. 16–30.
4. Hu B, Bi Y, Wu K, Fu R, Huang Z. A lightweight path validation scheme in software-defined networks. In: IEEE INFOCOM 2024—IEEE Conference on Computer Communications; 2024 May 20–23; Vancouver, BC, Canada: IEEE; 2024. p. 731–40.

5. Xu S, Zhao Y, Huang L, Qiao C. Routing and photon source provisioning in quantum key distribution networks. In: IEEE INFOCOM 2024—IEEE Conference on Computer Communications. Vancouver, BC, Canada: IEEE; 2024. p. 1411–20.
6. Nadimi-Shahraki MH, Zamani H, Varzaneh ZA, Sadiq AS, Mirjalili S. A systematic review of applying grey wolf optimizer, its variants, and its developments in different Internet of Things applications. Int Things. 2024;26:101135. doi:10.1016/j.iot.2024.101135.
7. Aslam N, Wang H, Aslam MF, Aamir M, Hadi MU. Intelligent wireless charging path optimization for critical nodes in internet of things-integrated renewable sensor networks. Sensors. 2024;24(22):7294. doi:10.3390/s24227294.
8. Zhang W, Geng H. Improving network availability through optimized multipath routing and incremental deployment strategies. Comput Mater Contin. 2024;80(1):427–48. doi:10.32604/cmc.2024.051871.
9. Liu D, Xu X, Liu M, Liu Y. Dynamic traffic classification algorithm and simulation of energy Internet of things based on machine learning. Neural Comput Applicat. 2021;33(9):3967–76. doi:10.1007/s00521-020-05457-7.
10. Lavate SH, Srivastava P. A hybrid feature selection approach based on random forest and particle swarm optimization for IoT network traffic analysis. Int J Electr Electro Res. 2023;11(2):568–74. doi:10.37391/ijeer.110244.
11. Gheni HQ, Oleiwi WK, Al-Barmani Z, Alabdali MA. Optimizing feature selection for intrusion detection: a hybrid approach using cuckoo search and particle swarm optimization. Int J Safety Secur Eng. 2024;14(6):1907–12. doi:10.18280/ijsse.140624.
12. Zhao R, Wang Y, Xue Z, Ohtsuki T, Adebisi B, Gui G. Semisupervised federated-learning-based intrusion detection method for Internet of Things. IEEE Int Things J. 2022;10(10):8645–57. doi:10.1109/JIOT.2022.3175918.
13. Hammi B, Zeadally S, Labiod H, Khatoun R, Begriche Y, Khoukhi L. A secure multipath reactive protocol for routing in IoT and HANETs. Ad Hoc Netw. 2020;103(15):102118. doi:10.1016/j.adhoc.2020.102118.
14. Raza S, Wang S, Ahmed M, Anwar MR, Mirza MA, Khan WU. Task offloading and resource allocation for IoV using 5G NR-V2X communication. IEEE Int Things J. 2021;9(13):10397–410. doi:10.1109/JIOT.2021.3121796.
15. Huang J, Wan J, Lv B, Ye Q, Chen Y. Joint computation offloading and resource allocation for edge-cloud collaboration in internet of vehicles via deep reinforcement learning. IEEE Syst J. 2023;17(2):2500–11. doi:10.1109/JSYST.2023.3249217.
16. Fan W, Su Y, Liu J, Li S, Huang W, Wu F, et al. Joint task offloading and resource allocation for vehicular edge computing based on V2I and V2V modes. IEEE Transact Intell Transport Syst. 2023;24(4):4277–92. doi:10.1109/TITS.2022.3230430.
17. Du Z, Liu F, Yuan W, Masouros C, Zhang Z, Xia S, et al. Integrated sensing and communications for V2I networks: dynamic predictive beamforming for extended vehicle targets. IEEE Transact Wirel Communicat. 2022;22(6):3612–27. doi:10.1109/TWC.2022.3219890.
18. Xu S, Sun J, Cao H, Gao Y, He Z, Wu C. Shield-U: safeguarding traffic sign recognition against perturbation attacks. In: 2024 IEEE 23rd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). Sanya, China: IEEE; 2024. p. 2453–61.
19. Gope P, Hwang T. Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks. IEEE Syst J. 2015;10(4):1370–9. doi:10.1109/JSYST.2015.2416396.
20. Gupta D, Kumar R. An improved genetic based routing protocol for VANETs. In: 2014 5th International Conference-Confluence the Next Generation Information Technology Summit (Confluence). Noida, India: IEEE; 2014. p. 347–53.
21. Zhao J, Ren R, Zou D, Zhang Q, Xu W. IoV-oriented integrated sensing, computation, and communication: system design and resource allocation. IEEE Trans Vehicular Technol. 2024;73(11):16283–94. doi:10.1109/TVT.2024.3422270.
22. Al-Hejri I, Azzedin F, Almuhammadi S, Syed NF. Enabling efficient data transmission in wireless sensor networks-based IoT applications. Comput Mater Contin. 2024;79(3):4197–218. doi:10.32604/cmc.2024.047117.
23. Khatua S, Mukherjee A, De D. FedGen: federated learning-based green edge computing for optimal route selection using genetic algorithm in internet of vehicular things. Veh Commun. 2024;12(2):100812. doi:10.1016/j.vehcom.2024.100812.

24. Park JH, Yang Q, Yoo SJ. RHRA-DRL: RSU-assisted hybrid road-aware routing using distributed reinforcement learning in internet of vehicles. IEEE Access. 2024;12(10):25385–96. doi:10.1109/ACCESS.2024.3366280.
25. Ma Z, Jiang J, Wei H, Wang B, Luo W, Luo H, et al. A blockchain-based secure distributed authentication scheme for internet of vehicles. IEEE Access. 2024;12(7):81471–82. doi:10.1109/ACCESS.2024.3409361.
26. Hasan MR, Zhao Y, Luo Y, Wang G, Winter RM. An effective AODV-based flooding detection and prevention for smart meter network. Procedia Comput Sci. 2018;129(4):454–60. doi:10.1016/j.procs.2018.03.024.
27. Kazmi SHA, Hassan R, Qamar F, Nisar K, Ibrahim AAA. Security concepts in emerging 6G communication: threats, countermeasures, authentication techniques and research directions. Symmetry. 2023;15(6):1147. doi:10.3390/sym15061147.
28. Amanlou S, Hasan MK, Bakar KAA. Lightweight and secure authentication scheme for IoT network based on publish-subscribe fog computing model. Comput Netw. 2021;199:108465. doi:10.1016/j.comnet.2021.108465.
29. 118E277 T. IoT traffic generation patterns dataset; 2021. [cited 2025 Mar 27]. Available from: https://www.kaggle.com/tubitak1001118e277/iot-traffic-generation-patterns.