Computer Systems Science & Engineering
DOI:10.32604/csse.2022.019557
Article

ML-Fresh: Novel Routing Protocol in Opportunistic Networks Using Machine Learning

Puneet Garg*, Ashutosh Dixit and Preeti Sethi

Department of Computer Engineering, J. C. Bose University of Science and Technology, YMCA, Faridabad, 121006, India
*Corresponding Author: Puneet Garg. Email: puneetgarg.er@gmail.com
Received: 17 April 2021; Accepted: 23 May 2021

Abstract: Opportunistic Networks (OppNets) is gaining popularity day-by-day due to their various applications in the real-life world. The two major reasons for its popularity are its suitability to be established without any requirement of additional infrastructure and the ability to tolerate long delays during data communication. Opportunistic Network is also considered as a descendant of Mobile *Ad hoc* Networks (Manets) and Wireless Sensor Networks (WSNs), therefore, it inherits most of the traits from both mentioned networking techniques. Apart from its popularity, Opportunistic Networks are also starting to face challenges nowadays to comply with the emerging issues of the large size of data to be communicated and blind forwarding of data among participating nodes in the network. These issues lower the overall performance of the network. Keeping this thing in mind, *ML-Fresh*-a novel framework has been proposed in this paper which focuses to overcome the issue of blind forwarding of data by maintaining an optimum path between any pair of participating nodes available in the OppNet using machine learning techniques viz. *pattern prediction, decision tree prediction, adamic-adar* method for complex networks. Apart from this, *ML-Fresh* also uses the history of successful encounters between a pair of communicating nodes for route prediction in the background. Simulation results prove that the ML-Fresh outperforms the existing framework of Opportunistic Networks on the grounds of standard Quality-of-Service (QoS) parameters.

Keywords: Opportunistic networks; artificial intelligence; machine learning; link prediction; routing protocols; QoS parameters

1 Introduction

Owing to the hard efforts put by network science researchers, man today can communicate even beyond the planetary boundaries. Due to the significant growth in the networking domain, data communication has been possible in such areas where it once seemed impossible. Opportunistic Networks are the result of such momentous growth in a couple of years in the field of wireless networking [1].

According to some researchers, Opportunistic Networks inherited their basic traits from Mobile *ad hoc* Networks (MANETs) [2] and Wireless Sensor Networks (WSNs) [3,4]. Yuan et al. [5] suggested that

Opportunistic Networks (OppNets) may be regarded as a part of Delay Tolerant Networks or Disruption Tolerant Networks. As the name itself indicates, this type of network is developed especially to tolerate long delays. There are various scenarios in real life like where this networking technique fits best. These scenarios may include Inter-Planetary communication, Disaster Management, War-Field situation, Space communications, etc.

Opportunistic Networks work on the principle of store-carry-and-forward i.e. the member node in the OppNet first stores the data which is supposed to be sent to some pre-decided node and send it to the desired location directly or through the intermediate node(s) [6]. The node stores the mentioned data until it gets a notification of the successful delivery to the destination. This feature makes the Opportunistic networks best suited for a situation like intermittent connection or connection failure.

According to Nayyar et al. [7], there is no standardized definition of Opportunistic Networks, but, based on Literature Survey, Opportunistic Networks may be described as a "network of wirelessly connected nodes". Opportunistic networks do not require any specific infrastructure support for their establishment. Following Garg et al. [8], An Opportunistic Network can be formed using smartphones via Bluetooth or Wi-Fi. Similarly, using the basic connection feature available in the device to be connected, Opportunistic Networks may form various kinds of networks such as Acoustic Networks, Airborne Networks, Mobile Communication, Vehicular Networks, etc. Besides the mentioned networks, Opportunistic networks may also include all types of mentioned networks to form a single hybrid network with the help of necessary infrastructure to connect heterogeneous network devices. A general scenario of such a situation is depicted in Fig. 1.

Apart from the above-mentioned traits, Opportunistic Networks has numerous applications in the real-life world; some of them are listed below:

a) Support for Smart City: Saloni et al. [9] developed LASSO which is an android based app that connects a group of people in a city via Bluetooth/Wi-Fi for communication and tracking each other. This app worked well on a group of tourists in an unknown city and preventing them from miss that place.

b) Under-water communication: Menon et al. [10] and Detweiller et al. [11] discussed the various methods to use Opportunistic Networks as a backbone networking technique for communicating above as well as underwater in sea region.

c) Forest surveillance: Martonosi [12] developed a project named Zebranet under the Princeton University project to monitor the routine life of Zebra in the Forest using Opportunistic Networks as the backbone. This project may be extended to cover all animals residing in the forest for surveillance purposes.

d) SWIM: Small et al. [13] elaborated an opportunistic network based on the concept of the Shared Wireless Info-station Model (SWIM) for data exchange over the long-range in ocean area. Initially, it was practiced by tying sensors on whales back and data was forwarded through the tied sensors and received successfully back on Info-Station.

e) Inter-Planetary Communication: Wood et al. [14] described Saratoga, an opportunistic network-based routing protocol specifically designed for inter-planetary data communication. Currently, Saratoga is used by NASA for few years.

f) Miscellaneous: Besides the above-mentioned applications, Opportunistic Networks have proved themselves highly beneficial in establishing Airborne Networks [15] and space operations [16].

1.1 Categories of Routing Protocols

Although Opportunistic Networks have numerous routing protocols with different strategies and working styles, to make it in a standardized format, all available protocols are categorized on the ground of some standard parameters such as working strategy [17], data forwarding techniques, etc. Juyal et al. [18] classified routing protocols of Opportunistic Networks into six major categories which are depicted in Fig. 2:

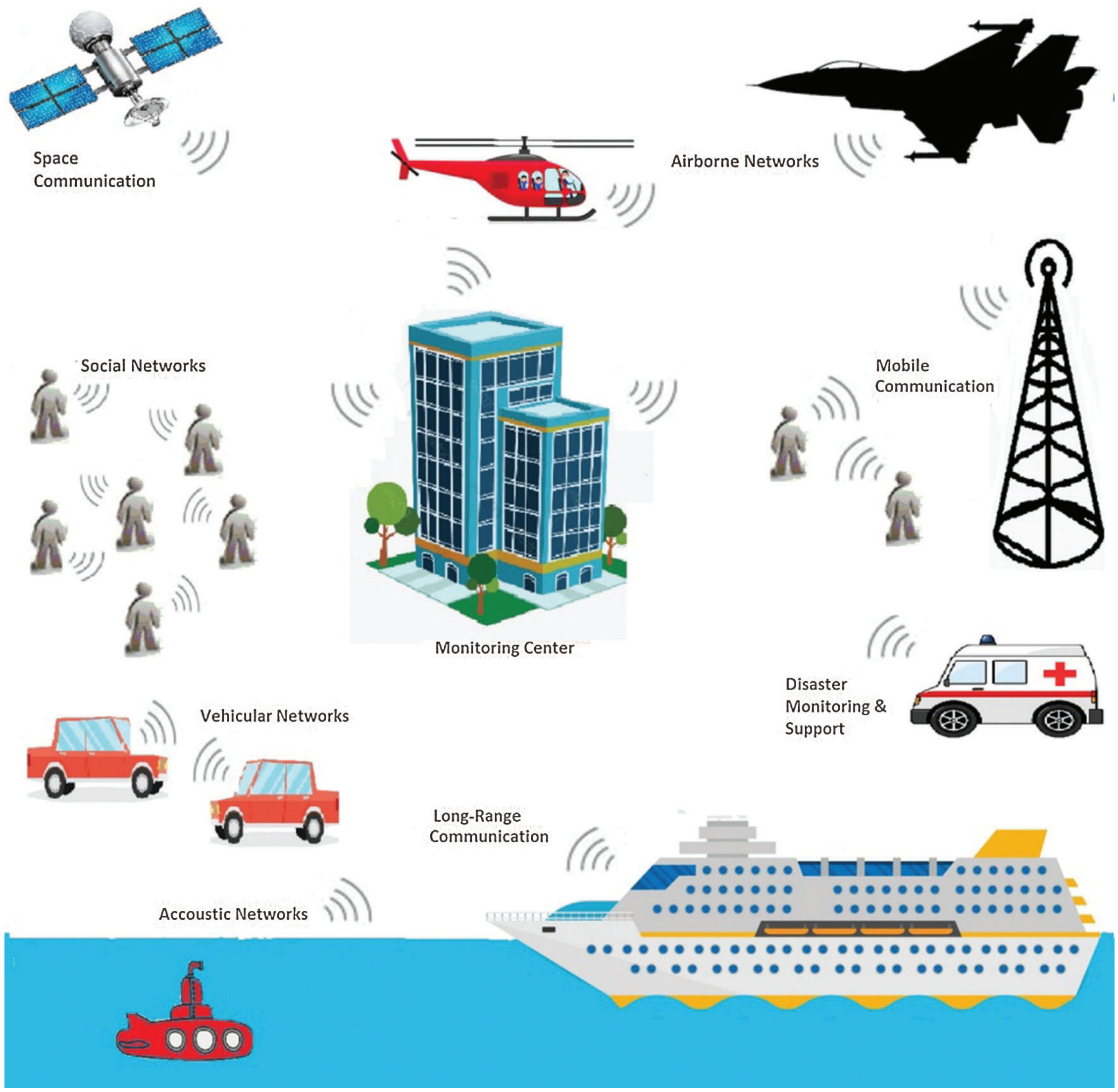

Figure 1: A general scenario of OppNet

1.2 Organization of Paper

This paper is categorized into five major sections. Section 1 is dedicated to the introduction of Opportunistic Networks (OppNets), their applications, and routing protocols. It primarily aims at the new researchers who do not have enough knowledge about the research work. The extensive research work in this arena is carried out in the next sections. Section 2 describes the motivation and recent research work that inspire authors towards the focused research. Section 3 explains the methodology and working strategy of the proposed work i.e., *ML-Fresh* followed by Section 4 which mentions the simulation and result in part of the research work carried out. Towards the end, Section 5 concludes the overall research work explained in this paper.

2 Motivation

Recent research work proved through simulation that *Fresh* Protocol from the category of social-community-based routing protocols delivers the best performance on the criteria of standard QoS parameters [19,20]. It motivates for appending this discussed research for further improvisation in the *Fresh* Routing framework to get the best out of it. But, before discussing further research, a basic somehow of *Fresh* Routing protocol is explained to get a better understanding of the proposed methodology of *ML-Fresh* Protocol in this paper.

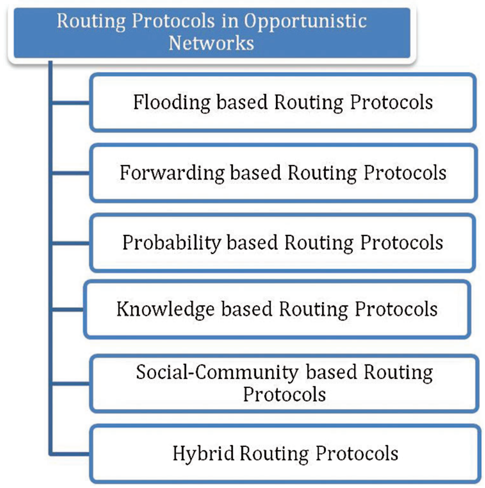

Figure 2: Classification of routing protocols in opportunistic networks

2.1 Fresh Protocol

Fresh (i.e., FResher Encounter SearcH) Protocol was introduced by Ferriere et al. [21]. Initially, it was designed for Mobile *Ad hoc* Networks but soon it was also adapted for Opportunistic Networks with few updates as per the network standards. Despite its simplicity, *Fresh* protocol is well known for its efficient route discovery in dynamic network topology which makes it a good choice for real-life situations. In *Fresh* Protocol, every node keeps a record of its encounter with other nodes present in the network in the process of route discovery along with the age gradient. Instead of searching new path for the destination node every time, the node firstly searches the destination node or any intermediate node towards the destination node from its record along with the respective age gradient. After the successful search, the node prefers the path or intermediate node which has the least encounter age i.e., which has more recent encounter experience with the destination node.

Some key points of *Fresh* Routing Protocol are as follows:

a) *Fresh* Protocol does not require any geographic knowledge about the participating nodes in the network as it follows the concept of Euclidean Space.

b) *Fresh* Protocol follows the concept of age gradient i.e. the path between source and destination node depends upon encounter history. It means the selected path may or may not be the shortest path between two nodes i.e. a straight line between source node to destination node on Euclidean Space.

c) Each Node in *Fresh* Protocol maintains its encounter history with all other nodes of the network; therefore, it does not require a common clock for synchronization.

d) Overall performance of *Fresh* Protocol does not get affected by the heterogeneous velocities of nodes, although it creates some hurdle during route discovery.

e) The search criteria of route discovery in *Fresh* protocol are Omni-directional.

2.2 Pseudocode

Pseudocode of *Fresh* Protocol is as follows:

```
proc FRESH (S,D) =
{
  if (thisnode.ID = D) then
  {
  replyToSource(S);
  }
  else
  {
  T := prevEncounterAge(D);
  A := findNextAnchor (D, T);
   if (A != D) then
   notifyNextAnchor(A, D);
  }
}
```

Fresh Protocol first searches the direct path between source node S and destination node D. If there is such path, it informs the source node otherwise it recursively searches intermediate node with the recent encounter with destination node D which can be used for data transmission from S towards D up to the point when data transmission is complete.

2.3 Literature Review

There have been various research attempts to make opportunistic routing better than the previous one. Some important efforts are listed below:

Sharma et al. [22] proposed a routing protocol named kROp(k-Means clustering based routing protocol for opportunistic networks) which uses machine learning techniques in the background for next-hop selection in route discovery during data dissemination. The output delivered by kROp was compared with PRoPHET and other standard opportunistic routing protocols. Results proved kROp as the better option in terms of hop count and delivery probability.

Lakshmi et al. [23] introduced a novel routing model known as SPR (Socialized Proficient Routing) for OppNet using Machine Learning. In this model, intermediate nodes are chosen on the ground of human-social traits to maintain the robustness of the path being selected for data forwarding. Various ML-based classifiers such as Decision-Tree, Neural-Networks, and Support-Vector-Machine are used during the training phase. Simulation results proved good performance using standard parameters.

Kara et al. [24] proposed an algorithm known as position-based hybrid routing algorithm (PBHRA) to exploit the bandwidth and energy consumption at the maximum level. It was implemented in MATLAB and compared with some standard Table-driven and position-based algorithms. The comparative study explained PBHRA outperforms the other algorithms taken.

Lambrinos et al. [25] presented a novel algorithm that is motivated by location-based Social Networks. Programmable controllers were used for route discovery and data transmission. A Portion of Machine learning is also used for training datasets containing the location-based information of nodes present in the Opportunistic network. Implementation of this algorithm delivered good performance on standard parameters over traditional location-based Opportunistic Routing Protocols.

Sharma et al. [26] developed a history-based routing protocol that uses the history of encounters between two nodes for current path selection in Opportunistic Networks using reinforced learning. To simulate this concept, a real mobility trace from INFOCOM-2006 was taken. Simulation results show that this strategy worked efficiently.

Souza et al. [27] introduced the "FriendShip and Acquaintanceship Forwarding" (FSF) protocol that performs data forwarding based on the Social relationship between the communicating nodes. It was implemented on ONE Simulator to inspect its performance on Message Delivery. The Proposed Protocol scored good results over standard metrics.

Rashidibajgan et al. [28] efficiently explained the advantages of using the history of node movement and interaction with other nodes during data communication. They proposed a Privacy-Preserving History-Based (PPHB) routing algorithm for implementing their proposition. In this approach, Each node participating in the opportunistic Network maintains a *History Table* (HT) which contains geographic knowledge about the other nodes encountered during data dissemination and every node decides after looking up its History Table for the next data communication.

Huang et al. [29] present a new framework based on Kernel regression named PreKR based on the Kernel Regression Technique for Link Prediction in Opportunistic Networks during route discovery. This research aims to enhance the packet delivery ratio by reducing the time consumption in route discovery in a network with dynamic topology. Experimental results showed that the proposed framework outperforms the standard method by 25% with 90% accuracy.

Sharma et al. [30] introduced a reinforced learning-based approach named RLProph for Opportunistic Networks. Its primary objective was to maximize message delivery probability using machine learning and reinforced learning in the background. With the help of simulation, it had been proved that RLProph delivered better performance over standard QoS parameters.

Li et al. [31] presented a new Link Prediction Method named *Combo-Pre* using a combination of machine learning and standard prediction methods. Experimental results proved its superiority over the traditional link prediction methods.

Janku et al. [32] presented the benefits of using unsupervised learning in Cluster Opportunistic Networks. The authors designed a hierarchical routing algorithm with the combination of three different strategies for decision making in route prediction using unsupervised learning techniques of machine learning. Experimental results show its better performance than traditional approaches.

The research in the above-mentioned research papers indicates that there are lots of opportunities to get a better result if machine learning is used with Opportunistic Networks. Keeping this thing in mind, this paper proposes the new routing protocol using Machine Learning in the background as well as adopting the Fresh Routing protocol as the backbone technique.

3 ML-Fresh

As discussed in previous sections of this paper, *Fresh* Protocol is observed as the best routing protocol among all the variety of routing protocols available in Opportunistic Networks [19]. Literature survey indicates the direction of incorporating machine learning in *Fresh* Protocol for better performance. Therefore, a new routing protocol known as *ML-Fresh* (Machine Learning enabled Fresh) Routing Protocol has been introduced.

Opportunistic Network involves blind forwarding of data to intermediate nodes to send it to the destination node, but, this blind forwarding deteriorates the overall performance of the Network due to unnecessary replication of single data which is supposed to be forwarded to a particular endpoint. This limitation can be removed by forwarding the data only to those nodes which are actual intermediate node during data transmission which can boost the delivery outcome by minimizing the delivery overhead of the whole network. This objective may be achieved by predicting the optimum suited path between source and destination node during data transmission.

ML-Fresh is based on such methodology. It uses the combination of three prime machine learning methods to predict the route in the dynamic topology of Opportunistic Networks with aim of achieving the best prediction from available options. The Three Machine Learning techniques are Pattern Prediction [33,34], Decision Tree Prediction [35,36], and Adamic-Adar Method [37]. Besides these, *ML-Fresh* also takes inputs from the history of successful encounters between two endpoints as Dhurandher et al. [38] proved it as a crucial factor for efficient routing in Opportunistic Networks.

The functioning of *ML-Fresh* protocol may be classified into the following phases:

a) Warm-up Phase
b) Preparation Phase
c) Prediction Phase
d) Decision Phase

It begins with the *Warm-up Phase* when participating nodes move as per the mobility model chosen and prepare their respective routing tables mentioning their recent encounter experience with other nodes. Like *Fresh* Protocol, *ML-Fresh* Protocol works on Euclidean space. The Formula for calculating Euclidean Distance (ED) [39] between each pair of nodes (a, b) in Set N is

$$ED = \sqrt{\sum_{i=1}^{n} (b_i - a_i)^2} \tag{1}$$

Besides saving recent encounter experiences with other nodes like *Fresh* Protocol, each node keeps the record of its every successful encounter with other nodes additionally in *ML-Fresh*.

The *preparation phase* starts with pattern mining of links between two nodes that are still connected as per the current timestamp t_i. It forwards such Links to the *Active Updated Links Set* block. It also transfers remaining Links for *Threshold Validity Checks* for validating their usage possibility for future use. After the Threshold check, such Links are categorized and forwarded into two categories. The first category contains those Links which were connected or active for some timestamp t_j but is not updated as time varies. The second category covers all those links set which were connected for a very few time or never connected. All Links set covered in three different groups are forwarded for further processing in the Prediction Phase as shown in Fig. 3.

Prediction Phase contains three different kinds of Link prediction mechanisms on three different kinds of Links set supplied. Firstly, Links set which are updated as well as active as per current timestamp are processed for pattern prediction for future use and forwarded to the *Decision Phase*. It involves the calculation of the smallest route between two nodes in the network which are not connected directly. Secondly, Decision Tree Prediction is used to process those links set which are not currently updated but were connected sometimes. The decision Tree-based prediction method [35] is one of the most popular Machine Learning-based methods involving the probability of successful communication. After the processing of Decision Tree Prediction, Links set with higher success probability are forwarded further. Lastly, Adamic-Adar Method is used to process those Links set which were connected for a very few time or never connected. Adamic-Adar Method is a very popular method among Machine Learning based Link prediction methods for Complex Networks and is regarded as the best one among the available options [37].

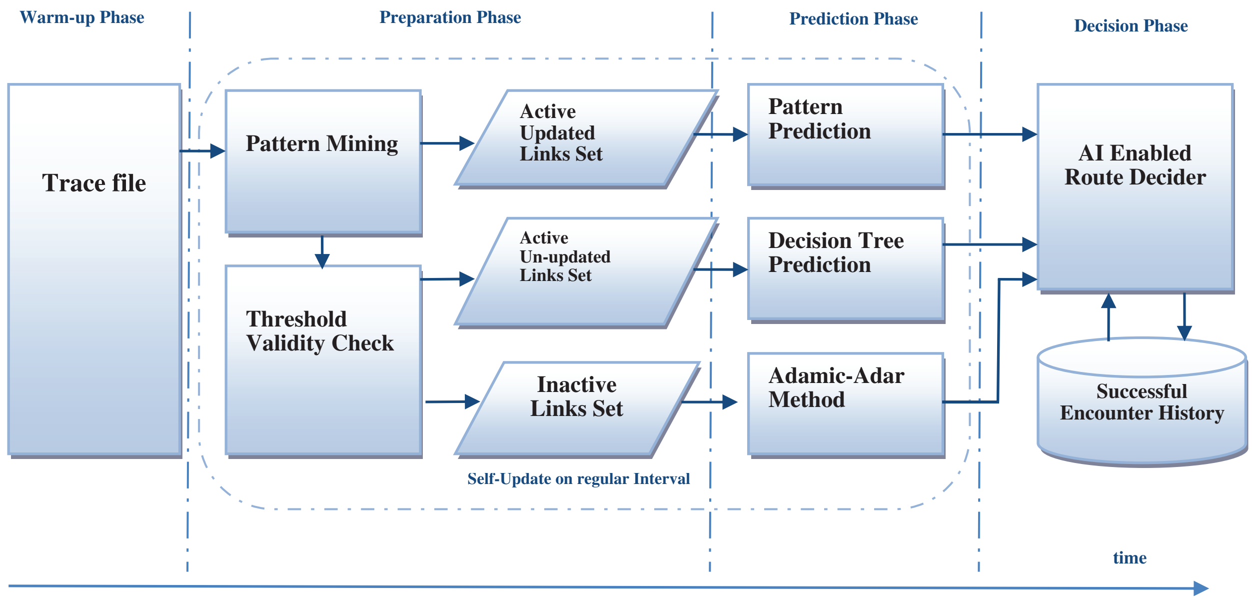

Figure 3: Framework of ML-fresh

Decision Phase receives three different Links set as Input to the *AI-Enabled Route Decider*. It also takes input from the History of successful encounters between two nodes. After receiving all inputs, *AI-Enabled Route Decider* optimizes and predicts a safe and secure route between two endpoints during data communication.

The working mechanism of *ML_Fresh* between the source node *S* and destination node *D* is shown through the pseudocode written below:

```
proc ML_FRESH (S,D) =
{
 if (thisnode.ID = D) then
 {
  replyToSource();
 }
  else
  {
  P [] := pattern_prediction(S,D);
  T [] := decision_tree(S,D);
  A [] := adamic_adar(S,D);
  H []:= encounter_history(S,D);
  R []:= optimum_route(P,T,A,H);
  }
}
```

ML-Fresh first searches whether the destination node *D* is in direct contact with source node *S* or not. If the two nodes are indirectly in contact, then, the communication starts directly through the If block otherwise four different functions viz. *pattern_prediction(S, D), decision_tree(S, D), adamic_adar(S, D) and* *encounter_history(S, D)* execute one by one. *pattern_prediction(S, D)* returns all active and updated intermediate node pairs between *S* and *D* to the array *P* using Machine Learning enabled Pattern Prediction mechanism. *decision_tree(S, D)* gives all active but un-updated intermediate node pairs between *S* and *D* to the array *T* using the probability-based decision tree method. *adamic_adar(S, D)* returns all intermediate nodes between *S* and *D* to array *A* using the Adamic-Adar method which is a prime technique for link prediction in complex networks. Moreover, *encounter_history(S, D)* returns the all node pairs between *S* and *D* from the history of successful encounters which involves routing between *S* and *D* to the array *H*. Lastly, all the intermediate node pairs obtained from four different functions described above are pass to *optimum_route(P, T, A, H)* through *P, T, A, H. optimum_route(P, T, A, H)* returns the exact and optimum path between S and D containing intermediate node pairs to the array *R* which can be retrieved for getting the optimized path.

4 Findings and Results

4.1 Simulation Environment

The performance of the *ML-Fresh* has been evaluated in the Opportunistic Network Environment (ONE) Simulator. This simulator has been chosen because, in accordance with Kuppusamy et al. [40], ONE is the most popular simulator used by researchers for Opportunistic Networks. ONE [41] is a simulator that provides visual output as well as trace files which can be analyzed through a plotting tool like Graphviz. The main advantage of using ONE Simulator over other simulators is that it easily generates event logs of even difficult mobility scenarios which are very much closer to the real-life situation which makes it the most suitable tool for applying in this research.

The performance delivered by the *ML-Fresh* is compared with the performance delivered by *Fresh* Routing Protocol in Opportunistic Network under common parameters listed in the next sub-section.

4.2 Common Parameters during Simulation

Some parameters that have been kept constant to achieve comparative performance between *Fresh* Protocol and *ML-Fresh* Protocol have been described in Tab. 1.

Table 1: Common parameters used during simulation

Parameters	Values
Simulation Area	12,000 x 10,000 Sq. Meters (120 Square KMs)
Simulation Time	1 Week
Movement Model	Cluster Movement
Time-To-Live (per Message)	1 Day (1440 Minutes)
Scenario Update Interval	0.1 Second
Communication Medium	Bluetooth, Wi-Fi (High Speed)
Bluetooth Interface Speed	250 Kbps
Bluetooth Interface Range	10 Meters
Bluetooth Interface Scan Interval	32 Seconds
Wi-Fi Interface Speed	500 Kbps
Wi-Fi Interface Range	10 Meters

(Continued)

Table 1 (continued).

Parameters	Values
Wi-Fi Interface Interval	64 Seconds
Node movement speed	From 0.5 m/s to 1.5 m/s
Transmission Range	10 Meters
Message Size	From 500 KB to 1 MB
Warm-Up period	1 Hour (3600 Seconds)
Operating System	The Mentioned research is carried out on MS Windows 10 Platform, but, the ONE simulator is a Java-based application, therefore, its performance is independent of the platform being used.
Initial Energy Provided	10 Lacs Units
Energy Consumed in Data Transmission	0.10 Units
Energy Consumed in Data Receiving	0.07 Units
Energy Consumed in Network Scanning for Data Communication	0.05 Units

4.3 Performance Metrics

Following Performance Metrics are chosen for the evaluation of *ML-Fresh* with *Fresh* Protocol:

a) Average Energy Consumption:

It is the energy consumed by the whole network on the average ground. The primary objective of every network researcher is to reduce it up to the maximum extent. It is inversely proportional to the network lifetime as the participating nodes have limited power in general.

Mathematically, it can be denoted as:

$$\boldsymbol{E_{Avg}} = \frac{\sum_{i=1}^{i=n} \boldsymbol{E_{Transmitted_i}} + \sum_{i=1}^{i=n} \boldsymbol{E_{Recieved_i}} + \sum_{i=1}^{i=n} \boldsymbol{E_{Scan_i}}}{\boldsymbol{n}} \tag{2}$$

b) Load-Delivery Ratio:

Load delivery ratio (LDR) is characterized as the proportion of data packets conveyed effectively to destination nodes and the total number of data packets produced for those destinations. LDR portrays the packet loss rate, which restricts the throughput of the network. The higher the delivery ratio, the better the performance of the routing protocol. LDR is determined as written below:

$$\mathrm{LDR} = \frac{\sum_{i=1}^{i=n} \mathrm{M}_{\mathrm{recieved}_i}}{\sum_{i=1}^{i=n} \mathrm{M}_{\mathrm{sent}_i}} \times 100 \tag{3}$$

c) Average Latency:

Average Latency is the average time consumed in the successful delivery of all messages transmitted between two endpoints during the complete network lifetime. High value of Latency is not considered desirable for a Good Network. The formula for its calculation is given below:

$$\mathrm{Latency}_{\mathrm{Avg}} = \frac{\sum_{i=1}^{i=n} T_{\mathrm{successful\ delivery}_i}}{\sum_{i=1}^{i=n} M_{\mathrm{Deliverd}_i}} \tag{4}$$

d) Error Rate:

Error Rate is the average number of lost messages during data transmission throughout the network lifetime. A good network has the least Error Rate. It can be calculated by the formula shown below:

$$\mathrm{Error\ Rate} = \frac{\sum_{i=1}^{i=n} \left(M_{\mathrm{created}_i} - M_{\mathrm{Successfully\ delivered}_i}\right)}{\sum_{i=1}^{i=n} M_{\mathrm{created}_i}} \tag{5}$$

4.4 Results

Following are the comparative analysis of the existing *Fresh* Protocol along with the *ML-Fresh* Protocol on the ground of mentioned parameters:

a) ***Average Energy Consumption:***

As depicted in Fig. 4, The *ML-Fresh* exhibits a good performance over *Fresh* Protocol on the criteria of Average Energy consumed during data communication. It gives the best performance i.e., 71% better than *Fresh* protocol when the number of nodes is 200. In average count, *ML-Fresh* outperforms *Fresh* Protocol by 11.65% on the ground of mentioned parameter.

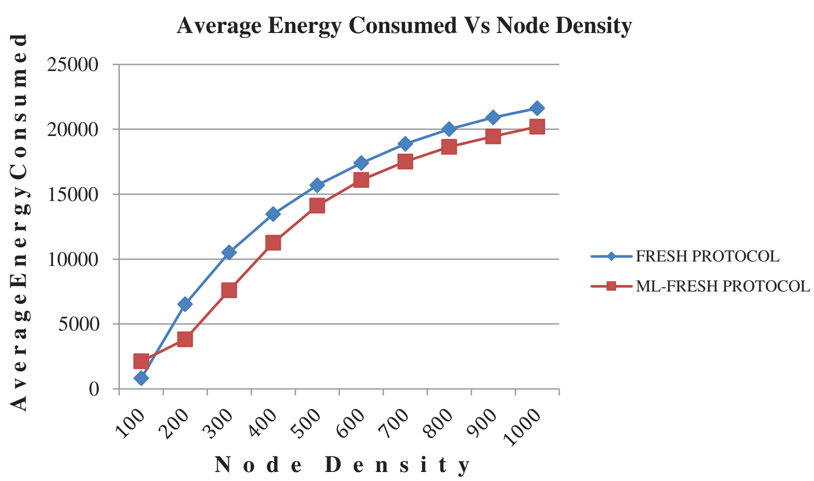

Figure 4: Protocols' performance based on Average Energy Consumption versus Number of Nodes

b) Load-Delivery Ratio:

The *ML-Fresh* proved itself beyond the expectations on the criteria of Load Delivery Ratio. Through Fig. 5, it was observed during simulation of both protocols along with the varying number of node densities that Load delivery Ratio (LDR) decreases as node density increases in the case of *Fresh* Protocol. It starts with LDR= 97.99 with node density of 100 and ends with LDR= 7.91 with a node density of 1000. On the contrary, LDR remains around the same i.e. 95.8 (Average) with all different cases of node densities in the case of *ML-Fresh*.

c) Average Latency:

After the simulation of both protocols with different Node Densities, It may be easily observed from Fig. 6, *Fresh* Protocol possesses more Latency than the *ML-Fresh*. However, The Latency difference between the two mentioned protocols decreases gradually as the node density increases. However, the *ML-Fresh* Protocol leads *Fresh* Protocol by 42% on average.

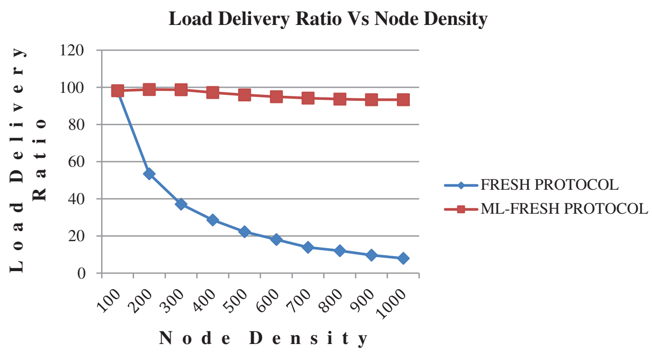

Figure 5: Protocols' performance based on Load Delivery Ratio versus Number of Nodes

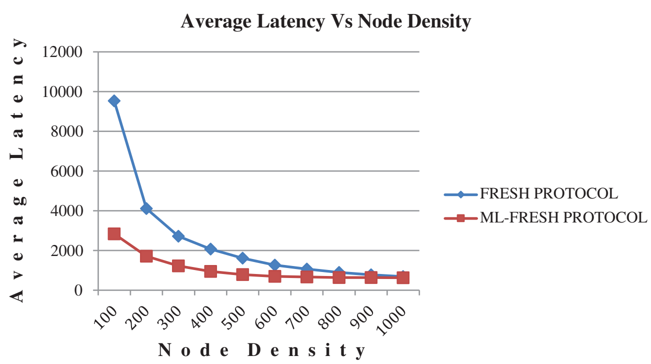

Figure 6: Protocols' performance based on Average Latency versus Number of Nodes

d) Error Rate:

Error Rate of any protocol decides the possibility of successful delivery of the desired data packet to the destined node. Lesser the Error Rate, communication will be more safe and secure. As Fig. 7 is showing the clear huge difference between the Error Rate of *Fresh* Protocol and *ML-Fresh* Protocol, It may be easily concluded that on average, *ML-Fresh* has 19 times less Error Rate than *Fresh* Protocol.

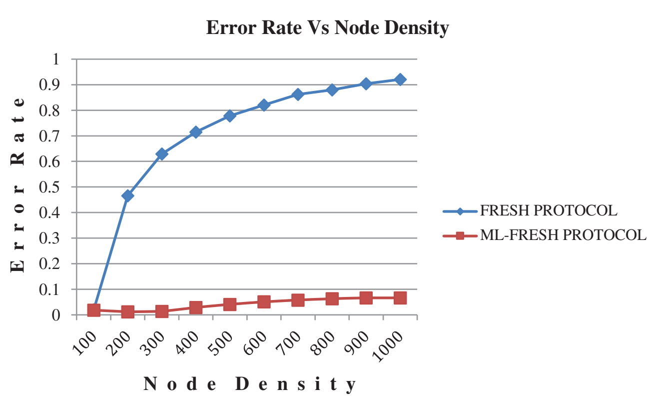

Figure 7: Protocols' performance based on Error Rate versus Number of Nodes

5 Conclusion

ML-Fresh protocol is inspired by *Fresh* Protocol which uses recent encounter experience for data transmission. *ML-Fresh* uses all three prime mechanisms from Machine Learning for Link Prediction in Opportunistic Networks. The simulation observations proved it as significant over *Fresh* Protocol. *ML-Fresh* delivered noteworthy performance on the ground of Average Energy Consumed, Load Delivery Ratio, Average Latency, and Error Rate. It is believed that *ML-Fresh* Protocol will motivate future researchers for further research.

Acknowledgement: The authors appreciate the J. C. Bose University of Science & Technology, YMCA, Faridabad, India, for providing research resources and equipment to encourage research among research scholars and faculty members.

Funding Statement: The authors received no specific funding for this study.

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

References

[1] N. Kaur and G. Mathur, "Opportunistic networks: A review," *IOSR Journal of Computer Engineering*, vol. 18, no. 2, pp. 20–26, 2016.

[2] N. S. S. Farheen and A. Jain, "Improved routing in manet with optimized multi path routing fine tuned with hybrid modeling," *Journal of King Saud University-Computer and Information Sciences*, pp. 1–8, 2020.

[3] P. Sethi, "Swarm intelligence for clustering in wireless sensor networks," in *Swarm Intelligence Optimization: Algorithms and Applications*. Hoboken, NJ, USA: Wiley, Chapter 13, pp. 263–273, 2020. https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119778868.ch13

[4] P. Garg, A. Dixit and P. Sethi, "Wireless sensor networks: an insight review," *International Journal of Advanced Science and Technology*, vol. 28, no. 15, pp. 612–627, 2019.

[5] P. Yuan, L. Fan, P. Liu and S. Tang, "Recent progress in routing protocols of mobile opportunistic networks: A clear taxonomy, analysis and evaluation," *Journal of Network and Computer Applications*, vol. 62, pp. 163–170, 2016.

[6] C. Prabha, S. Kumar and R. Khanna, "Analysis of routing and forwarding protocols in opportunistic networks," *Procedia Computer Science*, vol. 85, pp. 891–898, 2016.

[7] A. Nayyar, R. S. Batth, D. B. Ha and G. Sussendran, "Opportunistic networks: Present scenario-a mirror review," *International Journal of Communication Networks and Information Security*, vol. 10, no. 1, pp. 223–241, 2018.

[8] P. Garg, A. Dixit and P. Sethi, "Opportunistic networks: Protocols, applications & simulation trends," in *Proceedings of the 4th Int. Conf. on Innovative Computing and Communication (ICICC)*, New Delhi, India. 2021.

[9] M. Saloni, C. Julien, A. L. Murphy and G. P. Picco, "Lasso: A device-to-device group monitoring service for smart cities," in *Proc. of the International Smart Cities Conf. (ISC2)*, IEEE, pp. 1–6, 2017.

[10] G. Menon and P. M. J. Prathap, "Comparative analysis of opportunistic routing protocols for underwater acoustic sensor networks," in *Proc. of the Int. Conf. On Emerging Technological Trends (ICETT)*, pp. 1–5, 2016.

[11] C. Detweiller, I. Vasilescu and D. Rus, "An underwater sensor network with dual communications, sensing, and mobility," in *Proc. of the Oceans 2007*, Europe, pp. 1–6, 2007.

[12] M. Martonosi, *The princeton ZebraNet project: sensor networks for wildlife tracking*. Princeton University, pp. 2–7, 2004.

[13] T. Small and Z. J. Haas, "The shared wireless infostation model: A new ad hoc networking paradigm (or where there is a whale, there is a way)," in *Proc. of the 4th ACM Int. Sym. On Mobile Ad Hoc Networking & Computing*, pp. 233–244, 2003.

[14] L. Wood, W. M. Eddy, W. Ivancic, J. McKim and C. Jackson, "Saratoga: a delay-tolerant networking convergence layer with efficient link utilization," in *Proc. of the International Workshop on Satellite and Space Communications*, pp. 168–172, 2007.

[15] https://www.airbornewirelessnetwork.com/index. asp#advantages (Accessed on Februray 10, 2021).

[16] https://www.nasa.gov/ mission_pages/ station/ research/ experiments/ explorer/ Investigation. html?#id =717 (Accessed on Februray 10, 2021).

[17] M. Alajeely, R. Doss and A. Ahmad, "Routing protocols in opportunistic networks-a survey," *IETE Technical Review*, vol. 35, no. 4, pp. 369–387, 2018.

[18] V. Juyal, N. Pandey and R. Saggar, "Impact of varying buffer space for routing protocols in delay tolerant networks," in *Proc. of the International Conf. on Communication and Signal Processing (ICCSP)*, pp. 2152–2156, 2016.

[19] P. Garg, A. Dixit, P. Sethi and P. R. Pinheiro, "Impact of node density on the QoS parameters of routing protocols in opportunistic networks for smart spaces," *Mobile Information Systems*, vol. 2020, pp. 1–18, 2020.

[20] P. Garg, A. Dixit and P. Sethi, "Performance comparison of fresh and spray & wait protocol through one simulator," *Information Technology in Industry*, vol. 9, no. 2, pp. 452–457, 2021.

[21] H. Dubois-Ferriere, M. Grossglauser and M. Vetterli, "Age matters: efficient route discovery in mobile ad hoc networks using encounter ages," in *Proc. of the 4th ACM Int. Sym. On Mobile Ad Hoc Networking & Computing*, pp. 257–266, 2003.

[22] D. K. Sharma, S. K. Dhurandher, D. Agarwal and K. Arora, "kROp: k-Means clustering based routing protocol for opportunistic networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1289–1306, 2019.

[23] V. R. V. Lakshmi and G. K. Thonnuthodi, "Socialized proficient routing in opportunistic mobile network using machine learning techniques," *International Journal of Intelligent Engineering and Systems*, vol. 13, no. 4, pp. 434–445, 2020.

[24] R. Kara, I. Ozcelik and H. Ekiz, "A new routing algorithm in MANETs: Position based hybrid routing," *Scientific Research and Essays*, vol. 5, no. 3, pp. 328–338, 2010.

[25] L. Lambrinos and P. Kosmides, "Enhancing opportunistic networking using location based social networks," in *Proc. of the 8th ACM Int. Workshop on Hot Topics in Planet-scale mobile computing and online Social networking*, pp. 19–24, 2016.

[26] D. K. Sharma, S. Gupta, S. Malik and R. Kumar, "Latency-aware reinforced routing for opportunistic networks," *IET Communications*, vol. 14, no. 17, pp. 2981–2989, 2020.

[27] C. Souza, E. Mota, D. Soares, P. Manzoni, J. Cano *et al.,* "FSF: Applying machine learning techniques to data forwarding in socially selfish opportunistic networks," *Sensors*, vol. 19, no. 10, pp. 1–26, 2019.

[28] S. Rashidibajgan and R. Doss, "Privacy-preserving history-based routing in Opportunistic Networks," *Computers & Security*, vol. 84, pp. 244–255, 2019.

[29] D. Huang, S. Zhang, P. Hui and Z. Chen, "Link pattern prediction in opportunistic networks with kernel regression," in *Proc. of the 7th Int. Conf. on Communication Systems and Networks (COMSNETS)*, pp. 1–8, 2015.

[30] D. K. Sharma, J. J. P. C. Rodrigues, V. Vashishth, A. Khanna and A. Chhabra, "RLProph: A dynamic programming based reinforcement learning approach for optimal routing in opportunistic IoT networks," *Wireless Networks*, vol. 26, pp. 4319–4338, 2020.

[31] Y. Li and S. Zhang, "Combo-pre: A combination link prediction method in opportunistic networks," in *Proc. of the 24th Int. Conf. on Computer Communication and Networks (ICCCN)*, pp. 1–6, 2015.

[32] L. S. Janků and K. Hyniová, "Improvement of routing in opportunistic communication networks of vehicles by unsupervised machine learning," in *Proc. of the Int. Conf. on Engineering Applications of Neural Networks*, pp. 412–423, 2019.

[33] D. Huang, S. Zhang, P. Hui and Z. Chen, "Link pattern prediction in opportunistic networks with kernel regression," in *Proc. of the 7th Int. Conf. on Communication Systems and Networks (COMSNETS)*, pp. 1–8, 2015.

[34] X. Cai, J. Shu and M. Al-Kali, “Link prediction approach for opportunistic networks based on recurrent neural network,” *IEEE Access*, vol. 7, pp. 2017–2025, 2019.

[35] S. B. Chandanapalli, E. S. Reddy and D. R. Lakshmi, “DFTDT: Distributed functional tangent decision tree for aqua status prediction in wireless sensor networks,” *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 9, pp. 1419–1434, 2018.

[36] C. P. Koushik and P. Vetrivelan, “Heuristic relay-node selection in opportunistic network using RNN-LSTM based mobility prediction,” *Wireless Personal Communications*, vol. 114, pp. 2363–2388, 2020.

[37] A. Aleta, M. Tuninetti, D. Paolotti, Y. Moreno and M. Starnini, “Link prediction in multiplex networks via triadic closure,” *Physical Review Research*, vol. 2, no. 4, pp. 042029-1–042029-6, 2020.

[38] S. K. Dhurandher, D. K. Sharma, I. Woungang and S. Bhati, “HBPR: history based prediction for routing in infrastructure-less opportunistic networks,” in *Proc. of the 27th Int. Conf. on advanced information networking and applications (AINA)*, pp. 931–936, 2013.

[39] Z. Liao, L. Liu and Y. Chen, “A novel link prediction method for opportunistic networks based on random walk and a deep belief network,” *IEEE Access*, vol. 8, pp. 16236–16247, 2020.

[40] V. Kuppusamy, U. Miriya, A. Udugama and A. Förster, “Evaluating forwarding protocols in opportunistic networks: Trends, advances, challenges and best practices,” *Future Internet*, vol. 11, no. 5, pp. 1–26, 2019.

[41] A. Keränen, J. Ott and T. Kärkkäinen, “The ONE simulator for DTN protocol evaluation,” in *Proc. of the 2nd Int. Conf. On Simulation Tools And Techniques*, pp. 1–10, 2009.