A Robust Technique for Verifying the Neighbor Positions in MANET
Ms. Sejal Rana

1Department of Computer Science and Engineering

1Nar Narayan Shastri Institute of Technology, Jetalpur, Ahmedabad, Gujarat, India

Abstract— Mobile ad-hoc network is a collection of independent mobile nodes that can communicate to each other via radio waves. After nodes are deployed, they do not have knowledge about the neighbors so, they need to discover their neighbors in order to communicate with them. Knowledge of the neighbor nodes is essential for almost all routing protocols, medium-access control protocols and other topology control algorithms. Now, neighbor node may be trustable node or adversarial node so verification of node is required. For neighbor position verification NPV protocol is used. NPV protocol deals with verifying the neighbor node location. Nowadays, location awareness has become very important in MANET because wide range of protocol and application require knowledge of the position of participating nodes. So it is require improving NPV protocol. Examples of services that build on the availability of neighbor position information are Geographic routing in spontaneous networks, data gathering in sensor networks, movement coordination among robotics nodes, location specific services for handheld devices. Existing NPV protocol has some limitation like Lower Performance and False Positive Rate and False Negative Rate are high. So the objective of our proposed system is improve the FPR and FNR. Key words: Mobile Ad hoc networks, Neighbor position verification, False Positive Rate, False Negative Rate

I. INTRODUCTION

“MANET means Mobile ad hoc Networks. It is a collection of independent mobile nodes that can communicate to each other via radio waves” [1]. The mobile nodes that are in radio range of each other can directly communicate and those are out of range can indirectly communicate via intermediate or neighbor nodes to route their packets. After nodes are deployed, they do not have knowledge about the neighbors so, they need to discover their neighbors in order to communicate with them. Knowledge of the neighbor nodes is essential for almost all routing protocols, medium-access control protocols and other topology control algorithms. Now, neighbor node may be trustable node or adversarial node so verification of node is required. For neighbor position verification NPV protocol is used. NPV protocol deals with verifying the neighbor node location. Nowadays, location awareness has become very important in MANET because wide range of protocol and application require knowledge of the position of participating nodes. So it is required to improve NPV protocol. Examples of services that build on the availability of neighbor position information are Geographic routing in spontaneous networks, data gathering in sensor networks, movement coordination among robotics nodes, location specific services for handheld devices.

In this paper, the issues that are related to neighbor position verification (NPV) has been brought forth. In literature various methods are presented but there are no easy methods of having existing NPV recorded issues which are seen to be functional in an open wireless network surrounding without keeping faith on trusted nodes. The very recent remedy to the problem has been brought into light in. So one new set of rule to be represented in NPV which permit any wireless node in a mobile Ad Hoc network for continuously verifying the position of their nearer communication neighbor node without depending on priori faithfully nodes. But further this mechanism has also some drawbacks related to false positive and false negative rates underneath the occurrence of various attacks. Thus there is need to expand the existing algorithm and presented it in proposed system. And this expanded system only concentrates on improving the false positive and false negative rate. After that the survey on various methods to be included in the related work. In third part the proposed scheme and its proposed design is illustrated. Then at last the conclusion will be forecasted.

II. RELATED WORK

A. Node Discovery:

After nodes are deployed, they do not have knowledge about the neighbors so they need to discover their neighbors in order to communicate with them. Knowledge of the neighbors is essential for almost all routing protocols, medium-access control protocols and several other topology-control algorithms. Neighbor discovery is, therefore, a crucial first step in the process of self-organization of a wireless ad-hoc network [5].

The neighbors can be either physical neighbors or communication neighbor. The physical neighbors are those that are in the range of physical proximity of the discoverer. The communication neighbors are those that are reachable for communication but need not to be in the physical range of the discoverer.

1) Neighbor Discovery Algorithms:

a) Randomized neighbor discovery:

In randomized neighbor discovery, each node transmits at randomly chosen times and discovers all its neighbors by a given time with high probability.

b) Deterministic Neighbor Discovery:

In deterministic neighbor discovery, each node transmits according to a predetermined transmission schedule that allows it to discover all its neighbors by a given time with probability one.

c) Antenna based Neighbor Discovery:

The antenna models used in ad hoc networks are Omni directional antenna model or directional antenna model.

The Omni directional antenna model propagates signal in all directions. The algorithm used by Omni directional antenna is 1-way algorithm where the receiver will not send any acknowledgement after receiving the discovery message. The sender broadcasts the DISCOVER message to advertise itself. The receivers will discover one
neighbor if it receive the DISCOVER message correctly in the listen state. The Omni directional antennas have drawbacks like reduced gain, increased signal distraction, high bandwidth consumption, and increased noise.

Directional antennas model provide longer transmission range and higher data rate. They strongly reduce signal interferences in unnecessary directions and reduce jamming susceptibility. The directional antenna algorithms used to discover neighbors.

2) **Direct Discovery Algorithm:**
In direct discovery algorithm the nodes discover the neighbors which communicate with it directly. The method used to discover the neighbors is recording the angle of arrival of the beacon signal, determining the location based using GPS. The direct discovery algorithm will discover only those neighbors that communicate with it directly.

3) **Gossip based Algorithms:**
In gossip based algorithm the neighbors are discovered indirectly through the interaction with other neighbors. Messages are exchanged to discover the neighbors. The message consists of the list of neighbors’ IDs and their locations. The main drawbacks of gossip based algorithm are message length grows as more and more nodes are discovered and the presence of physical obstacles can cause nodes to incorrectly infer another node as its neighbor.

4) **Limitation of Neighbor Discovery Algorithms:**
These algorithms do not provide the security architecture to identify the malicious or misbehaving nodes. These algorithms do not verify the position of the neighbor nodes.

**B. Existing Methods:**

1) **Securely Determining Own Location:**
In MANET, self-localization is mainly achieved through Global Navigation Satellite Systems, for example GPS, whose security can be provided by cryptographic and non-cryptographic defense mechanisms [6].

2) **Secure Neighbor Discovery (SND):**
It deals with the identification of nodes which a communication link can be recognized or that are within a given distance [8]. SND is only a step toward the solution that simply place, an adversarial node could be securely discovered as neighbor and be indeed a neighbor (within some SND range), but it could still cheat about its position within the same range. In other words, SND is a subset of the NPV problem, since it lets a node assess whether another node is an actual neighbor but it does not verify the location.

In short SND just finds neighbor node but it does not verify the position of that node.

3) **Neighbor Position Verification:**
In [7], an NPN protocol is proposed that first lets nodes calculate distances to all neighbors, and then commends that all triplets of nodes encircling a pair of other nodes act as verifiers of the pair’s positions. This scheme does not rely on trustworthy nodes, but it is designed for static sensor networks, and requires lengthy multi-round computations involving several nodes that seek consensus on common neighbor verification.

In [9], an NPV protocol that allows nodes to validate the position of their neighbors through local observations only. This is performed by checking whether subsequent positions announced by one neighbor draw a movement over time that is physically possible. The approach in [9] forces a node to collect several data on its neighbor movements before a decision can be taken, making the solution unfit to situations where the location information is to be obtained and verified in a short time span. Moreover, an adversary can fool the protocol by simply announcing false positions that follow a realistic mobility pattern.

**C. NPV Algorithm:**
This algorithm basically used to discover and verify the position of its communication neighbors [10]. The below steps are used to explain the NPV algorithm.

- Step 1: discover nodes in range.
- Step 2: send request to nodes
- Step 3: wait for connection
- Step 4: get location from peers with time.
- Step 5: maintain location table
- Step 6: broadcast the location to other nodes
- Step 7: get response from other
- Step 8: verify the destination location and response from other nodes
- Step 9: check for location data at every request or operation
- Step 10: if the location of peer is invalid mark it as spam (by its mac id)
- Step 11: broadcast the spammed peer mac id to all other nodes.

Neighbor position verification is done through NPV algorithm.

---

Fig. 1: NPV in MANET
1) User Registration and Login for Ad Hoc Usage:
Every application needs to allow authorized user through authentication process. In this stage it’s used to create the ad hoc user for this application using both registration and login for ad hoc user screen. To avoid attackers in mobile ad hoc network this login and registration process is preliminary task to provide security. Ad hoc user registers their account in this application. Those who are already registered their account in this application; they can access their account through login.

2) Discover Own Location and Neighbor Location:
In this stage of process it’s used to find the own location and Neighbor location through the WIFI integrated service. These findings are used to involve in the neighbor position verification. This verification is done through the NPV algorithm. Secure transmission in mobile ad hoc network is complex and it’s achieved by NPV algorithm.

3) Connection between Neighbor Node:
Connection establishment with neighbor and accept connection by their neighbors made a connection more secure. In this stage it’s used to follow initial security mechanism through the cryptography techniques. Connections with their neighbors are established here using AES cryptography technique. Connection need to be accepted in both ends then only source can sent secure message transaction. Neighbor position verification algorithm used to check all with their neighbor through above mentioned steps to verify their neighbors.

4) Secure Content Transaction:
In final stage of this application implementation is secure content transaction to secure discovered neighbor destination. Position verification done through NPV algorithm and the message and attachments, whatever I need to send to the secure neighbor are happened to be here. Use send option after attachments and secure neighbor node selected.

5) NPV Protocol:
NPV means Neighbor position Verification Protocol. This Protocol exchanges messages and using verification test verify the position of communicating nodes [11].

Let a node S is called as a verifier, which discovers and verifies the position of its communicating neighbors. A verifier, S, can initiate the protocol at any time instant, by triggering the 4-step message exchange. The aim of the message exchange is to let S collect information it can use to compute distances between any pair of its communication neighbors. After the distances are calculated the nodes are classified as:
- Verified: Node is in the claimed position.
- Faulty: Node has announced an incorrect position.
- Unverifiable: Insufficient information.

The verification tests aim at avoiding false negatives (i.e., adversaries announcing fake positions that are deemed verified) and false positives (i.e., correct nodes whose positions are deemed faulty), as well as at minimizing the number of unverifiable nodes.

Here four set of messages are exchanged they are:
- POLL message
- REPLY message
- REVEAL message
- REPORT message

![Fig. 2: Message exchange](image)

A verifier S initiates this message. This message is anonymous. The identity of the verifier is kept hidden. Here software generated MAC address is used. This carries a public key K’S chosen from a pool of one time use keys of S’.

A communication neighbour X receiving the POLL message will broadcast REPLY message after a time interval with a freshly generated MAC address. This also internally saves the transmission time. This also contains some encrypted message with S public key (K’S). This message is called as commitment of X CX.

The REVEAL message is broadcasted using Verifier’s real MAC address. It contains A map MS, a proof that S is the author of the original POLL and the verifier identity, i.e., its certified public key and signature.

The REPORT carries X’s position, the transmission time of X’s REPLY, and the list of pairs of reception times and temporary identifiers referring to the REPLY broadcasts X received. The identifiers are obtained from the map MS included in the REVEAL message. Also, X discloses its own identity by including in the message its digital signature and certified public key.

6) Position Verification Test:
To verify the position of a node following three tests is done they are:
- Direct symmetry test
- Cross symmetry test
- The Multilateration Test

a) Direct symmetry Test:
In the Direct Symmetry Test, S verifies the direct links with its communication neighbors. To this end, it checks whether reciprocal Time of Flight-derived distances are consistent
with each other, with the position advertised by the neighbor, and with a proximity range $R$.

b) Cross symmetry Test:
In Cross symmetry test information mutually gathered by each pair of communication neighbors are checked. This ignores nodes already declared as faulty by the DST and only considers nodes that proved to be communication neighbors between each other, i.e., for which To F-derived mutual distances are available.

c) The Multilateration Test:
In Multilateration test, the unnotified links are tested. For each neighbor $X$ that did not notify about a link reported by another node $Y$, with $X, Y \in WS$ range. Once all couples of nodes have been checked, each node $X$ for which two or more unnotified links exist is considered as suspect.

Now take an example of topological information stored by verifier $S$ at the end of the message exchange and effect of a fake position announcement by $M$.

![Image](image_url)

**Fig. 3: Neighbor discovery in adversarial environment**

In Fig. 3, $S$ is verifier node which verifies the position of neighbor node. The neighbor nodes are $X$ and $Y$, $M$ is a malicious node announcing a false location $M'$, so as to fraudulently gain some advantage over other nodes. The figure portrays the actual network topology with black edges, while the modified topology, induced by the fake position announced by $M$, is shown with gray edges. It is evident that the displacement of $M$ to $M'$ causes its edges with the other nodes to rotate, which, in turn, forces edge lengths to change as well. The tests thus look for discrepancies in the node distance information to identify incorrect node positions. An independent knowledgeable adversary $M$ can move at most two links (with the verifier $S$ and with a shared neighbor $X$) without being detected: however, any additional link (e.g., with another shared neighbor $Y$) leads to inconsistencies between distances and positions that allow to identify the attacker.

7) *Dynamically Updating Neighbor Position:*
An ad hoc network is a collection of wireless mobile hosts forming a temporary network without the aid of any established infrastructure or centralized administration. In such an environment, it is necessary for one mobile host to enlist the aid of other hosts in forwarding a packet to its destination, due to the limited range of each mobile host’s wireless transmissions. In order to procure the position of other nodes while moving, an approach is proposed such a way that it helps in obtaining the position of a dynamic mobile node [11]. The neighbor discovery protocol is based on nodes sending notification messages tagged with ID and their current region whenever they enter or leave a region. In particular, there are three types of messages:

- Leave message
- Join message

- Join reply message

III. PROPOSED SYSTEM

There are various methods for verifying the neighbor position but there is not efficient or robust techniques which minimize the False Positive Rate and False Negative Rate. So we proposed a system which will minimize the FPR and FNR. For minimizing the FPR and FNR we included two parameter in the proposed system. These two parameters are Threshold Value and Time-out. This we will do by making some changes in the Algorithm of Message Exchange for verifier, Algorithm of Message Exchange for any neighbor, Algorithm of DST, Algorithm of CST and Algorithm of MLT.

![Image](image_url)

**Fig. 4: Proposed System design**
IV. CONCLUSION

In MANET a growing number of ad hoc networking protocols and location-aware services require that mobile nodes learn the position of their neighbors. So we require to verify the position of the neighbor node for secure communication. There are various methods for verifying the neighbor position but there are some limitation of Existing methods like lower performance and FPR & FNR are not minimize. So we proposed a robust technique which minimizes the FPR & FNR using Threshold and Time-out parameter.

REFERENCES