A Destination-Initiated Mix Route Algorithm in MANET

Dr. Pooja Sharma[1], Dr. Smriti Babbar,[2]

ILM Academy of Higher Learning- CET, Greater Noida[1], Shivaji college, University of Delhi[2]

Abstract: Traditionally, encryption has played an important role in network security. However, it is a misconception that to secure a network, one only needs to encrypt the traffic. Encryption protects communication partners from disclosure of their secret messages but cannot prevent traffic analysis and the leakage of information about who communicates with whom. In the presence of collaborating adversaries, it becomes very easy for them to reveal communication relations that can damage anonymity. However, reliably providing anonymity is crucial in many applications. In this paper, we explore the design of a MIX-net based anonymity system in mobile ad hoc networks. We focus on hiding the source-destination relationship with respect to each connection. We survey existing MIX route determination algorithms that do not account for dynamic network topology changes, which may result in high packet loss rate and large packet latency. We then introduce adaptive algorithms to overcome this problem. First one is Closest MIX Algorithm for End-to-End Connection and another one is Destination-Initiated MIX Route (DIMR) algorithm.

Keywords: Mobile ad-hoc network, Mix-based network, anonymity,

1. INTRODUCTION

Mix based networks [1] are based on the idea of mixing the incoming data traffic from one user with the data traffic coming from other user. Each mix has a public key which is used by the message senders to encrypt the message between the user and the mix. The mix accumulates these messages, decrypts them, optionally re-encrypts the messages and delivers them to the subsequent node. If there is a sufficient amount of input data packet from different sources, the mix ensures that the sources cannot be linked with the data packets once the packets come out of the mix. Users should not trust only one mix. Instead, they should send their data through a cascade of mixes. In this way, weak anonymity is preserved even if some of the mixes are honest (i.e., not run by an adversary).

The design of MIX-Net has been ported from its counterparts (e.g., Onion Routing [3] in wired networks. Firstly, the source node forward a data packet, called onion, and forward the routing information and key generation material to all MIXes that are available on the route. An onion is multi-layered data structures shown in Fig.1. Each layer contains information (e.g., the next hop MIX address, key seed, and expiration time) for one MIX. The first layer contains the information about the first MIX and the last layer holds the information about the last MIX on the route. At each step, a new layer is added and is encrypted with the public key of the MIX that is intended to decrypt the layer as shown in figure 1. When the node forward the packet to MIX A, MIX A receives the control packet called onion and other information such as the key seed material which is used for generating the decryption key. MIX A removes the one layer and sends the remainder of the onion to the next MIX on the route which is MIX B.

Since the source node knows all key seeds, it can generate the same keys as all MIXes, and use them for encrypting data packets. Mix A will decrypt the packet and forward it to the next MIX, and so on, until the last MIX tunnels the original packet to the destination node. For the security of onion we need a systematic approach. Either cryptography key system is used i.e., symmetric key cryptography or the public key cryptography, each data packet is encrypted with the keys of all MIXes on the route iteratively, similar to the construction of onion.

2. MIX-NET BASED ANONYMITY SYSTEM IN MANET

Mobile Ad hoc Network commonly called as MANET provides the technical platform for efficient information sharing in emergency and rescue operations without any centralized access point and infrastructure. The nodes move arbitrarily and its topology also changes frequently and unpredictability. The primary goal of an ad hoc network routing protocol is correct and efficient route establishment between a pair of nodes so that messages can be delivered in a timely manner. For ad hoc networks every node
communicates with other nodes in multi-hop manner through intermediate nodes directly or indirectly. Our goal is to provide connection anonymity, i.e., hiding the source-destination relationship of end-to-end connections established between mobile nodes. In order to achieve the same we use MIX-Net. We take an assumption that we deploy a large number of wireless MIX nodes in the environment in which the various mobile nodes already move around, the mobile nodes can create encrypted connections with the MIXes and therefore utilize the secure forwarding services that MIX nodes provide to achieve connection anonymity. A MIX node has limited transmission range just like the mobile node has limited transmission range. Therefore a mobile node takes the use of ad hoc routing mechanism in order to deliver packets to a MIX node. Similarly, data transfer between MIX nodes is also enabled by ad hoc networking. In a sense, the MIXes encompass an overlay network above the mobile ad hoc network. Figure 2 illustrates the hierarchical architecture of such a network. MIX-NET has the advantages of easy deployment, easy management, independence from underlying network, etc., but it also suffers from the performance problem. Because all the traffic goes through one or multiple MIXes, it may be possible that congestion can easily build up around MIX nodes which results in taking the long time to deliver the packet and a large number of packets can be drop during the transmission.

In a MANET, new problems arise which makes the situation worst. As we already explained, an ad-hoc network, the mobile nodes are often characterized by a dynamic topology due to the fact that nodes change their physical location by moving around. Assuming that if the topology of the network has changed and a connection still uses the same MIX route, then the packets can traverse through a long and detoured path to reach the destination node, which suffers a long delay and a higher loss rate. Based on this observation, we propose adaptive MIX route algorithms which allow data packets of each connection to take suitable MIX routes based on the current network topology.

In a MIX-based anonymity system, the selection of MIX route for each packet has direct impact on packet latency. When the source node forward the packet to its destination, the number of nodes that data packet has to travel to reach its destination is the sum of the lengths of the shortest paths between consecutive nodes on the MIX route, starting from the source node and ending with the destination node. It has been observed that an anonymized data packet consumes a lot of time for delivering as compared to an unanonymized data packet, which takes the shortest (or fastest) physical path for reaching to the destination. With regard to find or determine the MIX route, existing MIX-based anonymity systems fall into two cases. In one case, the single authority governed or administered all MIX nodes in a system [6][4]. In another case, all MIX nodes in a system are reachable from each other, i.e. fully connected [5][2]. In both cases, the MIX-net topology is known, remains unchanged or changes very infrequently. Based on this assumption, the following MIX route algorithms have been proposed and often used:

3.1 Approaches to Route Selection

a) MIX cascade [8]: The route can be constant, that is, it doesn’t change. This kind of mix network is usually referred to as “mix-cascade”. Although they are easier to implement and manage but these are much easier to traffic analyze. This algorithm returns a fixed, predetermined MIX route for all data packets. When using a MIX cascade, the degree of anonymity stays the same if all messages received by the first MIX are forwarded correctly to the last MIX. MIX cascade achieves the maximum security when at least one MIX is reliable. However, the degree of anonymity decreases when messages are lost on their way through the cascade (e.g. because of active attacks on the link between sender and first MIX of the cascade). If the eavesdropper access all MIXes in the cascade, then it access all the messages in the system and the security of the entire system will collapse.

b) Random MIX route [7][9]: The routes can also be chosen at random, that is, the user chooses i₁,i₂,…,iₙ uniformly at random. This type of mix network is usually referred to as “mix-net”. This algorithm returns a MIX route composed by randomly selected MIX nodes. A random route is constructed on a per-connection basis. Random MIX route eliminate the performance problem by distributing traffic loads among the MIX nodes. It is also resistant against active attacks in the sense that, if any MIX node becomes the victim of attacker; only messages whose MIX routes pass through the MIX are affected. This reduces the damage inflicted upon the system by compromised MIXes.

The above said approaches works effectively in static MIX-Nets but does not gives its maximum output in dynamic MIX-Nets. When the topology of the network has been changed then there will be a change in the length of the shortest paths between network nodes. If the fixed route of the network is used for transferring the data packet then it is quite possible, that the physical length of the route chosen becomes very large or there is a possibility of network congestion so the route becomes unavailable. Without referring to the network topology, if the random route is constructed, the route may
not be the optimal. In order to overcome the performance degradation, we try to develop a new MIX route algorithms which can works effectively when the network topology changes. These are Closest MIX Algorithm for End-to-End Connection and Destination-Initiated Mix Route Algorithm (DIMR).

4. CLOSEST MIX ALGORITHM FOR END-TO-END CONNECTION

In this section, we present a closest MIX algorithm. The purpose of the algorithm is to find closest MIX route for an end-to-end connection. We set several design goals for the algorithm. First, connection anonymity should not be violated during the route discovery process. Second, the algorithm should find a short Mix route based on the current network topology. As the network topology changes, the algorithm should update the Mix route. In this algorithm the source forward the data packet to its closest MIX node on the network route. We assume the distance between a mobile node and a MIX is the length of the shortest path between them. There will be the possibility that a mobile node has multiple closest MIX nodes that has the same distances which are connected to the mobile node. So each node maintains a Closest MIX List (CML). When the sources node wants to forward a data packet it can randomly select a MIX from its CML and forward the packet through the MIX which in turn forward to its destination. The algorithm is an extension of the Random Route algorithm where the MIX routes are find out on a per−packet basis. The data packets which have the similar source and the destination can take the different routes for delivering them and when the position of the node changes, its distance to each MIX may change as well. Thus we need a dynamic mechanism for updating a node’s CML in a timely manner. Therefore, we present two mechanisms: one is based on MIX solicitation anycasting and another is based on MIX advertisement flooding.

4.1 Mechanisms for Updating Closest Mix List

4.1.1 MIX Solicitation Anycasting Method

In MIX Solicitation Anycasting method, when a mobile node wants to forward its data packet, it can search its closest MIX node(s) by broadcasting a MIX Solicitation (MSOL) message over the entire network. The closest MIX nodes notify the mobile node by triggering the MIX Reply (MREP) message. A MSOL message contains the following parameters:

< MSOL, P, seqnum >

where P is the node who sends the message and seqnum is a unique sequence number generated by P. Initially, P forwards the message to their neighbor which is closest to it. If a mobile node receives the MSOL, it further transmits the message with the constraint that the same message is retransmitted only once within the transmission range. A simple technique for duplicate detection is to record the sequence numbers of all previously transmitted MADVs from different sources. But if any MIX node receives a MSOL, it does not retransmit the message, but unicasts a MREP message to the initiator of the search. In this sense, the MSOL is an anycast message. Each MIX only responds to the first MSOL it receives and ignores the further instances of the same MSOL.

MIX Reply (MREP) Message: A MREP message has the following parameter:

< M REP, M → P, seqnum >

where M represents the MIX node and seqnum which the mobile node copied from the MSOL message received from the P. When P receives all MREP messages, it can update its CML according to the information about the distance received from the network layer. The mobile node needs not to know about the closest MIXes when it does not have the packet to send to the intended node. So, it is clear that the MIX node has to be finding out on-demand. However, in an on-going connection, the initiator or the source node needs to find out its closest MIX frequently during the connection time in order to update its CML. In each new MIX discovery, the MSOL message carries a new sequence number. The broadcasting of a MSOL message in a network is illustrated in Figure 3. In this example, mobile node 6 finds over its MIX node by broadcasting the MSOL message, which is received by node A and node 7 immediately. Node A is a MIX node and node 7 is a mobile node. Since node A is a mobile node, so it does not retransmit the message, but replies a MREP message via unicast. Node 7 is a mobile node. So it retransmits the MSOL message, which is received by node 5 and node 8. Node A also receives node 7’s transmission but this is a duplicate.

4.1.2 MIX Advertisement Flooding Method

MIX advertisement is an alternative mechanism for MIX discovery. In this mechanism, each MIX periodically broadcasts MIX advertisement (MADV) messages, which are flooded over the network and propagate distance information to mobile nodes. A MADV message has the following parameters:

< M ADV, M, seqnum, hop_count >

where M is the MIX node who sends the message, seqnum is a unique sequence number generated by M. The combination
of M and the sequence number uniquely identify a MADV message. The hop count indicates how far the message receiver is from the message initiator. Initially, M sets the hop_count to 0. Whenever the mobile node retransmits the message, it increases the hop_count value by one. During the transmission, on the basis of received MADV messages, each mobile node maintains a list of Heard MIX nodes (HML). A mobile node only retransmits MADVs from the closest MIX nodes, i.e., MIXes in its CML. When a mobile node wants to retransmit a MADV, it faces some delays in forwarding the message called retransmission jitter, which is used to avoid collisions between the nodes. There may be a problem in this design. Take an example the mobile node which is located far away from the closest MIX may receive multiple copies of a MADV from the MIX, having the same seqnum value but different hop_count values, because the messages have been delivered through the different routes to reach to the destination mobile node. Due to retransmission jitter which is used for avoiding the collisions, it is quite possible that one instance or a copy of MADV with a larger hop_count value arrives earlier than another instance with a smaller hop_count value. In that case, the mobile node retransmits the MADV twice, because the later instance brings updated and “better” distance information.

For the reason of reducing the number of retransmissions made by mobile node due to the receiving of multiple instances of MADV, the prominent solution is that when the mobile node knows that the other instances of the same MADV can arrive frequently then the mobile node delays the retransmission of a received MADV. Because every MIX broadcasts MADVs periodically, there is a time for a mobile node to collect the data and calculate the length of time between arrival of the first and the arrival of the best instance of each MADV from each particular MIX, which is called settling time. Based on this data, a weighted average settling time of MADVs from a particular MIX is calculated by using the following formula, and helps to determine the length of delay before retransmitting a new MADV:

\[ ST_i = \alpha \times ST_i + (1 - \alpha) \times t_i \]

In the above formula, \( ST_i \) is the average settling time of MADVs from MIX i, \( t_i \) is the measured settling time of the last sequence number, and \( \alpha \) is a weighting parameter. In our simulations, \( \alpha = 0.875 \). There is a need for setting the time interval between two consecutive MADVs from each MIX. When the mobility of a node is high, the CML of each mobile node may change very frequently. In this case, a small MADV interval is desired, so that each mobile node has a time to updates its CML. When node mobility is low, the CML of each mobile node may remain stable for a longer time. A mobile node delays for \( 2 \times ST_i \) before retransmitting a MADV from MIX i, if i is in the CML; the initial value of \( ST_i \) is 1 second. In our simulations, we set the MADV interval to 5 seconds which is shown to be an appropriate value in moderately mobile networks.

When a mobile node has been located far away from a MIX, there will be the condition that the mobile node does not receive MADVs from the MIX in near future. There are two reasons behind this. First one, the mobile node has been temporarily disconnected from the MIX. Second reason, a neighbour of mobile node denies to retransmit the MADVs received from the MIX due to the availability of other closer MIX nodes. In either case, the MIX entry should be removed from the HML and CML. This can be achieved by setting expiration time for each entry. When MIX sends a new MADV, the entry’s expiration time will be reset. In our simulations, the expiration time of a MIX entry in HML and CML is 15 seconds. We can determine that lifetime of an entry is three times the MADV interval which is set by us as 5 seconds therefore, a node have the option to miss two consecutive MADVs from the MIX. If a MIX node receives a MADV then it does not retransmit the message and finally reject that message.

5. DESTINATION-INITIATED MIX ROUTE (DIMR) ALGORITHM

Here we discuss about the design of a new MIX route algorithm called as Destination Initiated MIX Route Algorithm. It is quite different from the Closest MIX algorithm. In the Closest Mix Algorithm the source node searches the MIX node closest to the source node regardless of the destination. But in the DIMR algorithm, the destination node of a connection initiates a route advertisement process, proactively, to propagate route information to the source node. DIMR algorithm has improved security because it can be combined with multiple MIXes in order to determine MIX routes. We now elaborate the MIX route discovery and maintenance process in the DIMR algorithm. It is divided into three phases:

1. Anonymous Route Request Phase: Whenever a source node wishes to communicate with a destination node, it initiates the route discovery process. Route discovery allows any node in the ad hoc network to dynamically discover a route to any other node, whether directly reachable within wireless transmission range or reachable through one or more intermediate network nodes. A node that initiates a route discovery, broadcasts a route request, which may be received by those nodes that lies within wireless transmission range of it. According to our algorithm, when a source node wants to send a data packet to the destination node through a MIX route, then before sending the packet, the source node sends an anonymous Route Request (RREQ) message to the destination node through a randomly chosen MIX route.

The random MIX route can be formed by MIXes selected from the node’s CML or HML. We can find out CML and HML information, by assuming that all MIXes broadcast MADV messages. The above can be explained by taking the example suppose Node S wishes to find a mix route for an anonymous connection destined to node D. S assembles a RREQ message and sends it to D via a custom mix route. As we mentioned, a custom mix route can be a random route
consisting of randomly chosen mixes, or be the closest mix of \( S \). The RREQ message is a unicast message. So \( S \) can encrypt the content of the message with \( D \)'s public key to prevent tracing of the message by an attacker. The RREQ packet may be lost during transmission. So a timeout-based retransmission mechanism must be activated by \( S \).

2. Node Registration Request Phase: When the destination node receives the RREQ message from the source node, then the destination node first selects a set of MIXes from its CML or HML and sends a Node Registration Requests (NRREQ) to each of them individually. Each Node Registration Requests holds a unique sequence number. The destination node sends the NRREQ that has the same sequence number to different MIXes at the same time.

3. Route Advertising Phase: The purpose of mix advertising for the mix nodes to announce their presence to non-mix nodes. Each non-mix node tries to pick the closest mix node as its first mix node on the route. Each MIX maintains a registration table, which lists all mobile nodes whose registration requests were recently received. Each registration table entry contains the following information about a particular node: The node’s address, the number of hops required to reach the node, the latest NRREQ sequence number.

The main emphasis of this algorithm is that each MIX in the network picks all the entries from its registration table and forward to its neighbours periodically which in turn dispatched throughout the whole network using the shortest path starting from the source MIX. Assuming that the network is connected, every node will receive the entries eventually. The path will consist of mix and mobile nodes through which the information will travel. There must be at least one MIX on the path, i.e., the source of the entries. From the path, a mobile node can find a MIX route to another mobile node that is currently registered at the MIX. This is why the process is called route advertisement. During a route advertisement, a MIX’s registration table entries are carried by a number of Route Advertisement (RADV) packets. Each RADV has the following parameters in it:

\[
< \text{RADV, M, sequum, mix_route, regtab} >
\]

where \( M \) is the MIX node who broadcast the advertisement, \( \text{sequum} \) is a unique sequence number generated by \( M \), \( \text{mix_route} \) is a sequence of MIX nodes through which the packet is delivered to the destination, and \( \text{regtab} \) is a copy of \( M \)’s registration table. Initially, the mix_route field contains one MIX, i.e., \( M \). When a mobile node wants to retransmits the packet, it does not make any changes in any field; but when a MIX node retransmits the packet, it appends itself to the end of mix_route. In a node’s route cache, each route is having a registration requests were recently received. Each registration table entry contains the following information about a particular node: The node’s address, the number of hops required to reach the node, the latest NRREQ sequence number.

As we know network is characterised by the fact that network topology may change over time, so it becomes very necessary to update the MIX route cache at each mobile node based on the current topology. As already mentioned in DIMR algorithm, the destination node starts a route advertisement process, to broadcast information to the source node. This algorithm defines that destination node sends registration request to a new set of MIXes nodes when it detects the changes in its CML or HML. Any MIX who receives a new NRREQ holding a new sequence number should update its registration table properly, and schedule a route advertisement.

The DIMR algorithm is an extension of CML algorithm. In this algorithm RREQ and NRREQ are delivered through CML. Both messages are unicast messages which generate small overhead in terms of route advertisement. Consider a network having of \( N \) mobile nodes and \( M \) MIXes, during a route advertisement interval, the MIXes generate \( M \) RADV messages and each RADV is flooded to all network nodes. Therefore, the total number of RADV transmissions is \( O((N + M)M) \).

5.1 Security Analysis

DIMR algorithm has improved security because it returns mix routes that consists of multiple mixes. In this paper our main target is to provide connection anonymity which can be achieved unless all MIXes on a route are compromised. Of course, the improved security is not achieved without cost. In the following, we focus on the topology attack that can be launched against the algorithm. We only consider attacks against connection anonymity.

Topology attack: In the DIMR algorithm, the source node receives MIX routes update from RADV messages which are flooded by the mix nodes in a network periodically. As the network topology changed, different mix routes has been used to deliver the packets in a long distance communication. If an eavesdropper gain the information about the network topology it becomes very easier for him to conduct an intersection attack in order to find out the destination node. It works as follows. If an eavesdropper has topology information, he will easily detect all MIX routes that a source node will use in order to deliver the data packets to each destination. Further he will easily find out to which the mix node the source node sends packets to and reducing the set of possible destinations. When the network topology changes and the source node uses a different MIX route for the same connection, an eavesdropper can derive a new set of possible destinations. The intersection of the two sets must contain the destination of the connection. In order to prevent from such attack, the prominent solution is that the source node sends the dummy or real packet to all the mix nodes and an eavesdropper cannot differentiate between real and dummy packet and gets no useful information. But it’s a costly solution. In order to reduce the cost the source nodes uses multiple MIX routes to send the data and each route starts with different MIX nodes.
This will complicate the attacker’s task and reduce the possibility of being compromised.

5.2 Simulation Model

The simulations are implemented using the ns-2 network simulation package [10]. The simulation environment contains 50 mobile nodes and a number of MIX nodes. We located all nodes in a 1000m by 1000m area arbitrarily. During the simulation MIX nodes do not change positions but mobile nodes frequently move in an area following the random way-point model of [11] with no pausing. When the simulation starts, each node moves toward its destination with a random constant speed. When a node reaches its destination, then it selects its new destination and starts to move towards it at a newly selected speed. The maximum node speed is a parameter for measuring node mobility. Each simulation is executed for 600 seconds. For our simulation we take a radio transmission range is approximately 250 meters. Each network interface transmits data at a rate of 2 Mbits /sec. The DSR protocol is used as the routing protocol. We implemented the Closest MIX algorithm and the DIMR algorithm above the ad hoc routing protocol. For making the comparison, we implemented a fixed MIX algorithm and a random MIX algorithm. In the fixed MIX algorithm, all the connections in the network use the same MIX in order to forward data packets, while in the random MIX algorithm, each connection uses a MIX randomly selected at the connection setup time. Each MIX node adds a random delay between 0 and 50 milliseconds before transmitting a data packet. Table 1 shows the values of parameters used in simulations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIX solicitation interval</td>
<td>5 secs</td>
</tr>
<tr>
<td>MIX advertisement interval</td>
<td>5 secs</td>
</tr>
<tr>
<td>MIX advertisement lifetime</td>
<td>15 secs</td>
</tr>
<tr>
<td>Route advertisement interval</td>
<td>15 secs</td>
</tr>
</tbody>
</table>

In all simulation runs, the source node inserts Continuous bit rate (CBR) traffic flows into the network. The size of data packet is 512 bytes. The source node generated the data packet at a rate of 4 packets per second. The source nodes and destination nodes of all connections are randomly chosen from the mobile node set. To evaluate the performance of a MIX route algorithm, we collect the following information:

a) Packet delivery fraction: The ratio of the data packets delivered to the destinations and those generated by sources.

b) Average end-to-end data packet latency: Average end to end delay includes all possible delays caused by buffering during route discovery latency.

c) Control overhead: This is the total number of control packets generated during the simulation run.

5.3 Simulation Results

In figures 4 to 6, we compare the performances of all MIX route algorithms as node mobility varies from 0 to 20 m/s. The number of connections is 10 and the number of MIX nodes is 5. Each value represents an average of 60 simulation runs. From Figure 4, we can see that all adaptive algorithms show significantly better performance than two non-adaptive algorithms.

![Fig.4: Packet latency vs. mobility](image)

![Fig.5: Total overhead vs. mobility](image)

Figure 5 shows that as the node mobility increases but the number of control packets generated by each adaptive algorithm does not increase. This explains the algorithm is not sensitive to node mobility. In the figure, it is shown that
MSOL generates larger number of control packets than MADV and DIMR. The reason for this is that MADV packets are flooded within the limited range within the whole network but each MSOL packet is flooded to all the nodes within a network. Although each RADV packet is also flooded to the entire network, the number of MIX nodes is typically lower than that of mobile nodes and the route advertisement interval is larger than MIX solicitation interval in our simulations. Figure 6 shows the distribution of control packets generated by the DIMR algorithm. Most of control packets are MADV and RADV packets. RREQ and NREG packets only make up a negligible portion of the total overhead.

![Figure 6: Distribution of overhead in DIMR vs. mobility](image)

6. CONCLUSION

In this paper, we explored the design of a MIX-network that provides anonymous connection services to mobile users in an ad hoc network. Our goal was to provide connection anonymity. In order to achieve the same, we used MIX-net. We proposed adaptive MIX route algorithms which allow data packets of each connection to take suitable MIX routes based on the current network topology. We explored two adaptive MIX route algorithms. First one is Closest MIX Algorithm for End-to-End Connection and another one is Destination-Initiated MIX Route (DIMR) algorithm. Compared to the Closest MIX algorithm, the DIMR algorithm has improved security because DIMR algorithm has improved security because it can be combined with multiple MIXes in order to determine MIX routes. We conducted extensive simulations to evaluate the performance of the proposed algorithms. It is shown that the adaptive algorithms achieve significant performance gain than nonadaptive algorithms in terms of packet delivery fraction and packet latency.

REFERENCES