Journal of Universal Computer Science, vol. 15, no. 5 (2009), 1090-1118
submitted: 1/8/08, accepted: 28/2/09, appeared: 1/3/09 © J.UCS

Cooperation Enforcement in a Highly Dynamic Mobile Ad Hoc Network

Yao H. Ho
(School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 326816-2362, USA
yho@cs.ucf.edu)

Ai Hua Ho
(School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 326816-2362, USA
aho@cs.ucf.edu)

Kien A. Hua
(School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 326816-2362, USA
kienhua@cs.ucf.edu)

Fei Xie
(School of Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 326816-2362, USA
xiefei@cs.ucf.edu)

Abstract: Operations of mobile ad hoc networks rely on the collaboration of participating nodes to route data for each other. This standard approach using a fixed set of nodes for each communication link cannot cope with high mobility due to a high frequency of link breaks. A recent approach based on virtual routers has been proposed to address this problem. In this new environment, virtual routers are used for forwarding data. The functionality of each virtual router is provided by the mobile devices currently within its spatial proximity. Since these routers do not move, the communication links are much more robust compared to those of the conventional techniques. In this paper, we investigate techniques to enforce collaboration among mobile devices by identify and punish misbehaving users in supporting the virtual router functionality. Simulation results based on various system configurations are given. They indicate that the proposed technique is effective.

Keywords: Cooperation-enforcement, mobile ad hoc networks, selfishness.
Categories: C.2.2. C.4

1 Introduction

Mobile Ad hoc NETworks (MANETs) have attracted great research interest in recent years. A mobile ad hoc network is a self-organizing multi-hop wireless network where all hosts (often called nodes) participate in the routing and data forwarding process. The deployment of ad hoc networks does not rely on fixed infrastructures such as router and base station, thereby posing a critical requirement on the nodes to cooperate with each other for successful data transmission. Many works (e.g., [Buchegger, 02], [Buttyan, 00], and [Jiang, 05]) have pointed out that the impact of malicious and selfish users must be carefully investigated. Existing cooperation enforcement techniques ([Buchegger, 02], [Buttyan, 00], [Jiang, 05], [Karp, 00], [Marti, 00], and [Michiardi, 02]) cannot be adapted for some of recent advance in routing protocols. In particular, we are interested in the new Connectionless-Oriented Approach [Ho, 04] and [Fubler, 03]. We investigate two such techniques, namely *Connectionless Approach* (CLA) [Ho, 04] and *Contention-Based Forwarding* (CBF) [Fubler, 03], in this paper. These techniques do not maintain a hop-by-hop route for a communication session to minimize the occurrence of broken link. In CLA, the network area is divided into non-overlapping grid cells, each serving as a *virtual router*. Any physical router (i.e., mobile host), currently inside a virtual router, can help forward the data packet to the next virtual router along the virtual link. This process is repeated until the packet reaches its final destination. Since a virtual link is based on virtual routers which do not move, it is much more robust than physical link. Another scheme, CBF, simply forwards data packets to the next hop without first having to establish the one-hop connection. The nodes that happen to be in the general direction towards the destination node help to forward the data packets.

The goal of this research is to address the cooperation issue for connectionless-oriented approach (i.e., CBF [Fubler, 03] and CLA [Ho, 04]) in wireless ad hoc networks. There can be both selfish and malicious nodes in a mobile ad hoc network. The selfish nodes are most concerned about their energy consumption and intentionally drop packets to save power. The purpose of malicious node is to attack network using various intrusive techniques. In general, nodes in an ad hoc network can exhibit Byzantine behaviors. That is, they can drop, modify, or misroute data packets. As a result, the availability and robustness of the networks are severely compromised. Many works ([Buchegger, 02], [Buttyan, 00], [Jiang, 05], [Karp, 00], [Marti, 00], and [Michiardi, 02]) have been published to combat such problem - misbehaving nodes are detected and a routing algorithm is employed to avoid and penalize misbehaving nodes. These techniques, however, cannot be applied to the connectionless-oriented approach since any node in the general direction towards the destination node can potentially help forward the data packets.

The primary contributions of this paper are as follows:

- We introduce a cooperation enforcement technique, called 3CE (*3-Counter Enforcement*), for the connectionless-oriented approach.
- We apply the 3CE method to two connectionless-oriented techniques:
  - Connectionless Approach (CLA), and
  - Contention-Based Forwarding (CBF).

- We present simulation results to show that with the 3CE features, CLA and CBF can prevent malicious nodes and enforce the cooperation among nodes to maintain the good performance of the network.

The remainder of this paper is organized as follows. We review the different connectionless-oriented protocols and different cooperation enforcement techniques in Section 2. We discuss the node configuration and present our cooperation enforcement techniques for connectionless-oriented techniques in Section 3. We give simulation results in Section 4 to demonstrate the benefits of the proposed techniques. Finally, we draw conclusion on this work in Section 5.

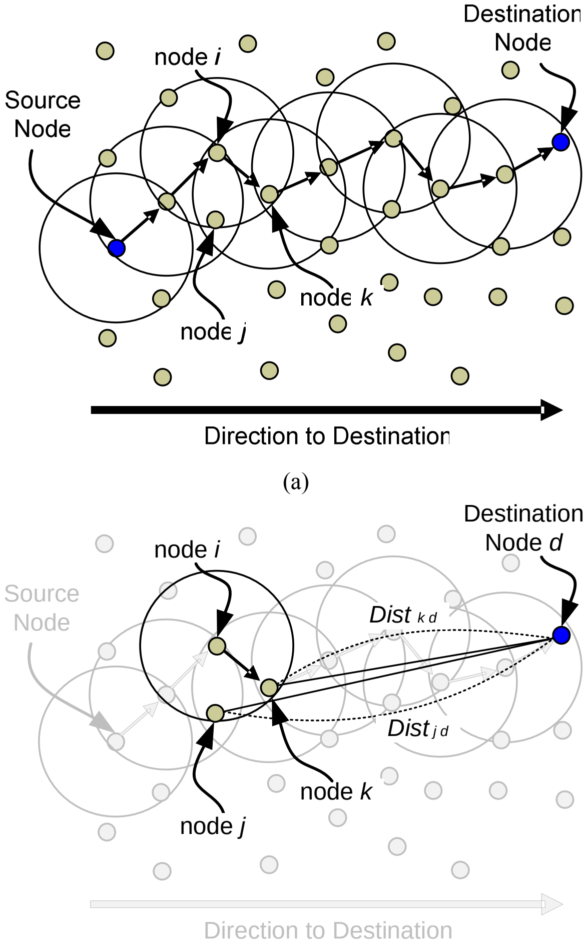
2 Related Work

In this section, we first briefly describe two connectionless-oriented techniques, namely CLA and CBF. We then review existing collaboration enforcement methods.

2.1 Connectionless-Oriented Routing Protocols

A good routing algorithm for MANETs must adapt to traffic patterns with minimal overhead. To reduce network flooding, routing protocols can leverage location information obtained from GPS (Global Positioning System) or other location services. For instance, LAR [Ko, 98] uses location information to limit the area of flooding, subsequently reducing route request messages. These schemes result in better power conservation and improve network scalability. To address mobility issues, hop-by-hop approaches, such as TBF [Niculescu, 03], TMNR [Blazevic, 05], and GPSR [Karp, 00], have been proposed. In those approaches, the node only needs to establish the connection to the next hop before forwarding the data. To determine the next hop, a node compares the distances of its neighbor nodes to the destination node (i.e., GPSR), the next waypoint (i.e., TMNR), or a trajectory (i.e., TBF).

CBF [Fubler, 03] and CLA [Ho, 04] are more recent routing techniques for MANETs. While GPRS, TMNR, and TBF need to establish a connection to the next hop before forwarding a data packet, CBF and CLA simply forward data packets to the next hop without first having to establish the one-hop connection. We classify these schemes as *Connectionless-Oriented Approach*. In CBF, a node forwards the packets as a single-hop broadcast to all neighbors. The neighbors compete with each other for the "right" to forward the packet. During this contention period, a node determines how well it itself is suited as a next hop for the packet. The node that wins the contention *suppresses* the other nodes, thus establishes itself as the next forwarding node. This contention is based on the distance of the nodes to the destination (see Figure 1(a)). For example, if node *j* and node *k* received a data packet from node *i* (see Figure 1(b)). Both node *j* and node *k* will calculate a contention (e.g., delay) timer according to their respective distances, $Dist_{j\,d}$ and $Dist_{k\,d}$, to destination node *d*. In this case, node *k*'s timer expires first (i.e., $Dist_{j\,d} < Dist_{k\,d}$) and broadcasts the data packet to the next node. This will cancel node *j*'s timer to prevent multiple next hops and packet duplication.



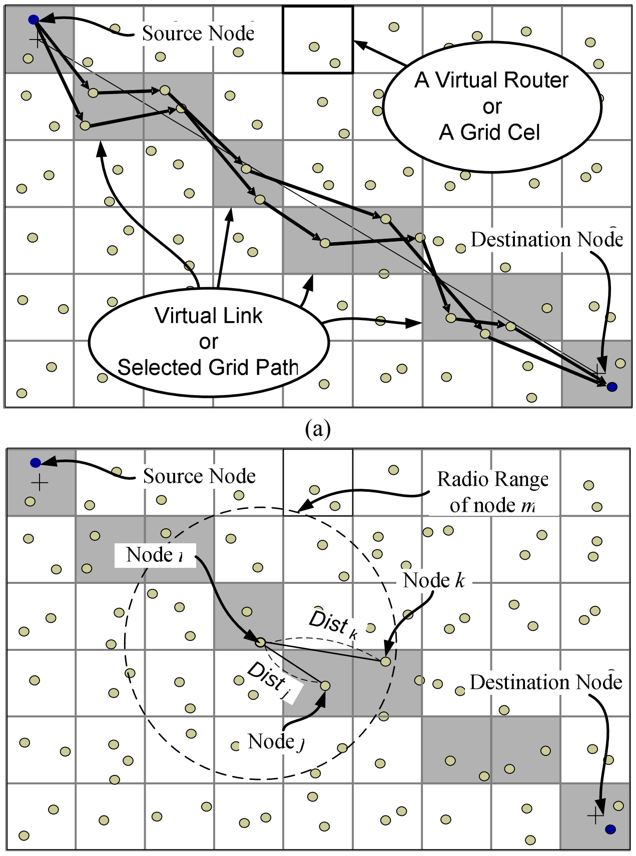
(a)

(b)

Figure 1. Contention-Based Forwarding.

In CLA ([Ho, 04] and [Ho, 07]), the network area is divided into small non-overlapping grid cells (see Figure 1(a)). Instead of maintaining a hop-by-hop route between the source and destination node, the source selects a list of grid cells that form a “connecting” path between the source and destination. From a different perspective, each grid cell can be viewed as a *virtual router* in the sense that any physical router (i.e., a mobile node) currently within the virtual router can alternate in forwarding data toward the next virtual router. The communication path consisting of consecutive virtual routers form a *virtual link* (see Figure 1(a)). For example, if node *j* and node *k* received a data packet from node *i* (see Figure 1(b)). Both node *j* and node *k* are within the radio range of the sender, node *i*. However, node *k* is farther away, and therefore has a shorter forwarding delay. As a result, node *k* will forward the data packet from node *i*. In this paper, we use the terms “virtual router” and “grid cell" interchangeably. Similarly, we also use the terms "virtual link" and "forwarding grid path" interchangeably.

Given a virtual router, its physical routers compete to forward the data packets according to a data forwarding procedure. This function computes a shorter delay for a node farther from the sender and closer to the destination. In this environment, a virtual link is considered broken if one of its virtual routers becomes empty. This is addressed by replacing the empty virtual router with a neighboring virtual router. The fundamental advantages of CLA are twofold. First, a virtual link is much less likely to become broken than a standard route used in conventional connection-oriented techniques; and second, unlike standard routes, the robustness of virtual link is not sensitive to the mobility inherent in MANET. Since both CBF and CLA can robustly support high mobility, these two schemes have been adapted for vehicle-to-vehicle environment.



(a)

(b)

Figure 2. Connectionless Approach.

2.2 Cooperation Enforcement Techniques

There are several Cooperation Enforcement Techniques to deter misbehaving nodes. We introduce some of them in this section. In Zhou and Hass [Zhou, 99], authors employ asynchronous threshold security and share refreshing for distributed certification authorities for key management in mobile ad-hoc networks. They take advantage of inherent redundancies in mobile ad hoc networks given by multiple routes to enable diversity coding, allowing for Byzantine failures give by several corrupted node or collusions. The approach is a potentially strong prevention mechanism; however, to the best of our knowledge, the impact on performance of a large scale network and ability to adapt to high mobility has not been published.

Smith, Murthy, and Garcia-Luna-Aceves [Smith 97] examine the routing security of distance vector protocols in general and develop countermeasures for vulnerabilities by protecting both routing messages and routing update. They propose sequence numbers and digital signatures for both routing messages and updates. However, distance vector protocols are not suitable to large scale and high mobility network as studied in CBF [Fubler, 03] and CLA [Ho, 04]. And it is difficult to employ [Smith 97] in such a network environment and to adapt to new types of routing protocols.

Buttyan and Hubaux propose incentives to corporation by means of so-called nuggets [Buttyan, 00]. Nuggets serve as a per-hop payment in every packet or counter to the secure module in each node to encourage forwarding. Similar approach called Confidant protocol proposed by Buchegger and Le Boudec [Buchegger, 02] which propagates the bad reputation of node to more than one node. However, this type of approaches cannot be employed by connectionless-oriented approach for the following reasons: First, malicious nodes can easily cheat the proposed protocols by creating and forwarding packets to a none-existing node or a random node to increase the nuggets or the counters since there is no pre-determined route or a next hop in connectionless-oriented approach. Second, in large scale networks, the connections between two nodes can have large number of hops. Thus, to establish a connection might be very costly or not affordable to some nodes in terms of nuggets.

Marti, Giuli, Lai, and Baker [Marti, 00] observe increased throughput in mobile ad-hoc network by complementing DSR with a watchdog (for detection of malicious behavior) and a ‘pathrater’ (for trust management and routing policy, every path used is rated), which enable nodes to avoid malicious nodes in their routes. Their approach does not punish malicious nodes that do not cooperate, but rather relieves them from the burden of forwarding for other. In other words, the malicious nodes are rewarded in their behavior. Jiang, Sheu, Hua, and Ozyer [Jiang, 05] proposed a finite-state model to penalize the misbehavior nodes and allow them to rejoin only if the behavior improved. However, in Connectionless Approach (CLA) and Contention-Based Forwarding (CBF), there is no pre-determining next hop. Thus, it is impossible to employ a misbehavior detection mechanism (i.e., watchdog) and a malicious node avoidance routing protocol (i.e., path rater).

Yi, Naldburg, and Kravets [Yi, 01] propose a modification of AODV with security metrics to path computation and selection. They define trust levels according to organizational hierarchies with a shared key for each level, so that nodes can state their security requirements when requesting a route and only nodes that meet these requirements can participate in the routing. Again, it is not suitable to connectionless based approach due to no per-determined route or selection process for a route.

3 3-Counters Enforcement (3CE) for Collaboration in Connectionless-Oriented Protocols

In this section, we first briefly describe the configuration of mobile nodes and their Tamper Proof Module. We then present our cooperation enforcement techniques, called 3CE, for connectionless-oriented techniques.

3.1 Node Configuration and Tamper Proof Module

The proposed technique is based on nodes with the following configuration. First, nodes are equipped with wireless interface cards that can be switched to detection mode to "detect" data transmission on a "suspicious" node in their proximities. Second, connectionless-oriented routing protocol is employed in the network layer. Without loss of generality, we base our discussion on the more recent techniques developed for routing in MANETs (i.e., Connectionless Approach routing protocol (CLA) [Ho, 04] and Contention-Based Forwarding (CBF) [Fubler, 03]). Nevertheless, the technique can be incorporated into any location-aid protocols to protect nodes against uncooperative behaviors. Third, reliable communication protocols such as TCP cannot be employed in this type of routing protocols. While TMNR and TBF need to maintain (proactively or reactively) neighbor nodes location information and establish a connection to the next hop before forwarding a data packet, CBF and CLA simply forward data packet without first establishing the link to the next node. Any node that happens to be in the general direction towards the destination node can compete for the "right" to forward data packets.

In addition, similar to the techniques presented in [Buttyan, 00] and [Jiang, 05], we also equip each node with a tamper resistant module. All other hardware and software components are susceptible to illicit modifications. We notice that a tamper-proof security module remains controversial [Pfitzmann, 97], but it proves to be inevitable in a large scale and high mobility network environment. Our approach guarantees that as long as the tamper resistant module is not compromised, nodes cannot benefit from uncooperative behaviors. Some mission critical data is stored in the tamper resistant module. This information include: 1) a unique *ID* of the node; 2) a pair of public/private keys; 3) a ***Forward Request Counter*** that counts number of packets that are received and need to be forwarded; 4) a ***Forward Counter*** that counts number of packets have been forwarded; 5) a ***Location Discovery Counter*** that counts number of Location Discovery packets initiated by a node; 6) a ***Session Table*** that keeps track ongoing communication sessions; 7) a **Counter Update Procedure** that updates the three counters; 8) a **Misbehavior Detection Procedure** that initiates the detection to identify a malicious node. Since the tamper proof module maintains information of three counters that are used to determine maliciousness of a node and initiate the detection, hereafter we also refer to this module as the 3C Module, and the proposed technique as the 3CE or 3C Enforcement technique.

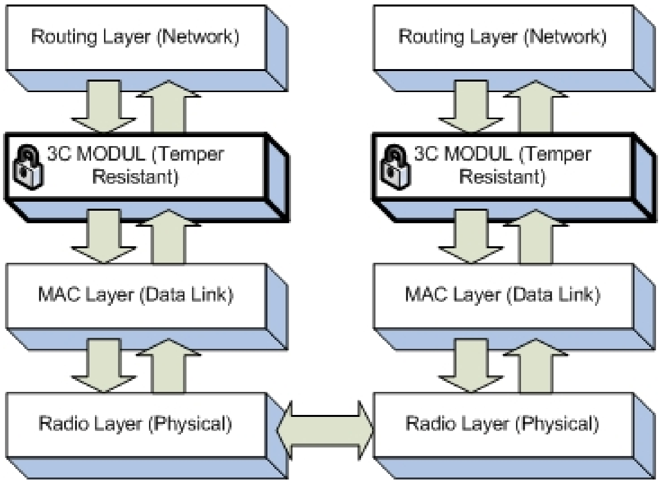


Figure 3. Layer Structure.

The 3C Module inspects Location Discovery packets, Location Reply packets and data packets exchange between the network layer and the MAC layer (see Figure 3); and the module updates the counters as follows:

1. When a new packet arrives at a non-destination node, it updates (i.e., increment by one) its ***Forward Request Counter***;
2. When a node forward a packet, it updates (i.e., increment by one) its ***Forward Counter***; and
3. When a note initiates a Location Discovery packet, it updates (i.e., increment by one) it's ***Location Discovery Counter***.

In addition, the 3C Module constructs and adds 3C's header (i.e., the value of three counters) to the Location Discovery packet as in various layers of the OSI model.

3.2 3C Module

In a connection-oriented (i.e., hop by hop route) approach, before a node can start a data transmission session to another node, the protocol needs to issue a route request to find a route to the destination node. However, in connection***less***-oriented approach, only the location of the destination node is needed. Thus, a Location Discovery packet is broadcasted to find only the destination's location. Once its location is determined, intermediate nodes can forward data packet according to the general direction towards the destination; and all packets exchanged between nodes are examined by the nodes' 3C Module.

In a 3C Module, three counters (i.e., ***Forward Request Counter***, ***Forward Counter***, and ***Location Discovery Counter*****)** are updated according to the counter update procedure. These counters are maintained by the node's own 3C Module (see Figure 3). Similar to [Buttyan, 00] and [Jiang, 05], we assume the 3C Module is a tamper resistant module that malicious users cannot contaminate it. The details of the counters update procedure will be discussed in Section 3.3 and 3.4.

When a source node *S* initiates a Location Discovery packet, node *S*'s 3C Module adds the 3C's header to the Location Discovery packet as in various layers of the OSI model. **3C header** contains the value of three counters (i.e., ***Forward Request Counter***, ***Forward Counter***, and ***Location Discovery Counter***) of node *S*. Based on this header, neighboring nodes of *S* can decide to forward or discard the Location Discovery packet. If a node *n* "suspects" the source node *S* is misbehaved, *n* invokes its **Misbehavior Detection Procedure**. A node suspects another node is misbehaving if one of the following is true: a) the *Forward Ratio* (i.e., ratio of *Forward Counter* to *Forward Request Counter*) of *S* falls below the *Forward Ratio* of *n*; or b) the *Request Ratio* (i.e., ratio of the *Location Discovery Counter* to *Forward Counter*) of *S* rises above the *Request Ratio* of *n*. If so, *n* exchanges 3C information (i.e., the value of the three counters) with its neighboring nodes to determine the network condition in the local area (i.e., *n*'s neighboring nodes). If the source node *S* is identified (by **Misbehavior Detection Procedure**) as misbehaving, its neighboring nodes will penalize this node by not forwarding *S*'s Location Discovery packets.

In order for malicious nodes to rejoin the network, non-malicious nodes still allow malicious nodes to participate in forwarding data. Unlike many techniques that avoid the malicious nodes during the routing procedure, our approach allows malicious nodes to rejoin the network by contributing its share (i.e., forwarding data for others) of network workload. This way, nodes are given more incentive to act collaboratively. By forwarding data packets for other nodes, a malicious node can increase its ***Forward Counter***. When its ratio of ***Forward Request Counter*** to ***Forward Counter*** rises above threshold α and its ratio of ***Location Discovery Counter*** to ***Forward Counter*** fells below threshold β, the malicious node will again be allowed to join the network, i.e., its neighboring nodes again help forward its Location Discovery packets. We elaborate the above processes in the following sections.

3.3 Counters Update during the Location Discovery Phase

As mentioned earlier, a node needs to find the location of the destination before it can start to send data packets in connectionless-oriented protocols such as CBF and CLA. A node can initiate a Location Discovery procedure, receive a Location Discovery packet, or forward/reply a Location Discovery packet. To initiate a Location Discovery procedure, a source node broadcasts a Location Discovery packet.

Location Discovery packet: Location Discovery packet contains the following information: source node ID (*source_ID*), source node's location (*S_cell_ID*), destination node ID (*destination_ID*), destination node's location (*D_cell_ID*), forward node ID (*forward_ID*), and forward node's location (*F_cell_ID*).

When a node receives a Location Discovery packet, it checks if it is the destination node. If so, it returns a Location Reply packet that contains its location (*D_cell_ID*); otherwise, if the node did not see this Location Discovery packet before, it adds its ID and its cell ID (i.e., forward node ID - *forward_ID* and the currently location - *F_cell_ID*) and broadcasts the Location Discovery packet to other nodes. In Figure 4, we show the data forwarding procedure for CLA in Routing Layer. The same procedure can be applied to CBF.

Session Table: Each node maintains a *Session Table* in its 3C Module to track all the ongoing communication session. An ongoing communication session is identified by a *session_ID* which is a pair of *source_ID* and *destination_ID* of the communication session. This table contains the following information for each entry (i.e., communication session): *session_ID* (i.e., a pair of *source_ID* and *destination_ID*) and a *time to live* (*TTL*) *timer*. An entry is deleted from the *Session Table* when one of the following information is true: (i) A communication session ended; (ii) Entry's *TTL* (time to live) *timer* expired; (iii) Entry belongs to an identified malicious node. An entry's *TTL timer* is reset when a packet received such that: *a*) the packet corresponds to this entry (i.e., *source_ID* and *destination_ID = session_ID*) and; *b*) it is not from a malicious node.

3.3.1 Initiate Location Discovery

When a **Location Discovery procedure** in the routing layer passes an initiated Location Discovery packet to the 3C Module, it processes the packet and updates the ***Location Discover Counter*** as follows (see Figure 4):

1. The 3C Module determines if this Location Discovery packet belongs to one of the initiator's (i.e., the source node's) ongoing communication session in the *Session Table*. If it does not belong to an ongoing session, go to Step 2; otherwise, go to Step 3.
2. The 3C Module increments the ***Location Discovery Counter*** by one and adds it to the *Session Table* (and go to Step 3).
3. The 3C Module adds a 3C header containing the values of the three counters (i.e., ***Forward Request Counter***, ***Forward Counter***, and ***Location Discovery Counter***) to this Location Discovery packet before passing it to the MAC Layer for broadcast to other nodes.

In the connectionless-oriented approach, the destination of a communication session is periodically updated according to the mobility of the destination node. The location of the source node is updated by piggybacking the location information in the data packets. However, a source node sometime needs to re-discover the location of a destination node due to packet losses caused by congestion, mobility, or channel errors. Thus, we differentiate between the initial location discovery and the location discovery that is re-establishing an ongoing communication session.

3.3.2 Receive Location Discovery Packet

When a Location Discovery packet broadcast from a node *m* to any of its one-hop neighbor node *n*, *n*'s MAC Layer passes the packet to its 3C Module for processing the Location Discovery packet and updating the ***Forward Request Counter*** as follow (see Figure 4):

1. The 3C Module determines if *m* is the source node that initiated this Location Discovery packet (i.e., packet's *source_ID = forward_ID*). If so, go to Step 2; otherwise, go to Step 3.
2. If *m* is the source node of this Location Discovery packet, the 3C Module in *n* uses the information in the packet's 3C header to determine if there is a need to start the detection procedure to examine *m*'s behavior. We will discuss when to initiate the misbehavior detection and the procedure for misbehavior detection in Section 3.5 and 3.6, respectively. If node *m* is confirmed to be misbehaving, the

3C Module of node *n* discards the packet (as punishment); otherwise, go to Step 3.

3. Node *n* keeps records of ongoing communication session in its *Session Table*. If the arriving Location Discovery packet's *source_ID* and *destination_ID* match an entry in node *n*'s *Session Table* (e.g., packet's *source_ID* + *destination_ID* = *session_ID*), its 3C Module resets the *time to live* (TTL) timer of the corresponding entry. Next, the Location Discovery packet is then passed on to the routing layer (Step 5).
4. If the Location Discovery packet is not belonged to any ongoing session in the *Session Table* (e.g., packet's *source_ID* and *destination_ID* ≠ *session_ID*), the 3C Module updates the *Session Table* and increases the ***Forward Request Counter*** by one. The 3C Module then passes the Location Discovery packet to the routing layer for further processing (Step 5).
5. Depending on different routing protocols (e.g., *CLA* or *CBF* protocol), node *n* can discard the packet, continue to forward (i.e., pass back down to lower layers), or initiate a reply procedure (i.e., reach the destination). In Figure 4, we show the routing protocol for CLA in the Routing Layer.

3.3.3 Forward or Reply Location Discovery Packet

Depending on the role of a node in a communication session (e.g., forwarding node or destination node), a node can forward the Location Discovery packet, reply to the Location Discovery packet with a Location Reply packet, or discard the Location Discovery packet according to its routing protocol. A Location Reply packet is generated by a node's Routing Layer when a Location Discovery packet arrived at a destination. This destination node needs to reply the source node of the Location Discovery packet. If a node is the destination, its Routing Layer generates a Location Reply packet and passes this reply packet to 3C Module.

When Routing Layer submits a Location Discovery packet or a Location Reply packet to 3C Module, 3C Module processes the packet and updates the ***Forward Counter*** as follows:

1. 3C Module determines if the Location Discovery packet or the Location Reply packet matches an entry in the *Session Table*. To determine if the Location Reply packet matches an entry in the *Session Table*, 3C Module simply reverses the order of *source_ID* and *destination_ID* of this packet. If the packet matches an entry in the *Session Table*, go to Step 2. Else, the packet is discarded because a malicious node can generate dummy packets to increase its ***Forward Counter*** to avoid detection.
2. 3C Module increases the ***Forward Counter*** by one. Then, the Location Discovery packet or the Location Reply packet is passed to MAC Layer.

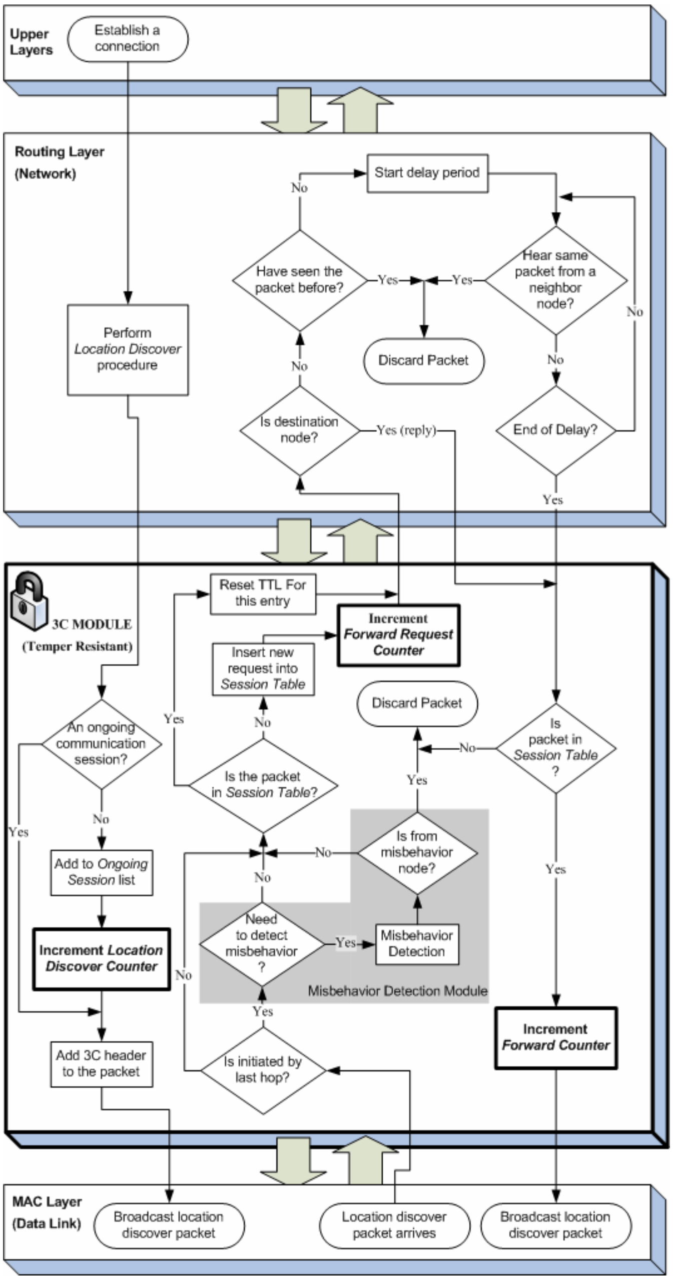

Upper Layers
Establish a connection
Routing Layer (Network)
Start delay period
No
Have seen the packet before?
Yes
Yes
Hear same packet from a neighbor node?
No
Perform Location Discover procedure
Discard Packet
No
No
Is destination node?
Yes (reply)
End of Delay?
Yes
3C MODULE (Temper Resistant)
Reset TTL For this entry
Increment Forward Request Counter
Insert new request into Session Table
Discard Packet
Yes
No
No
An ongoing communication session?
Is packet in Session Table ?
Is the packet in Session Table?
Yes
Yes
No
Is from misbehavior node?
Add to Ongoing Session list
No
No
Yes
Need to detect misbehavior ?
Yes
Misbehavior Detection
Increment Location Discover Counter
No
Misbehavior Detection Module
Yes
Increment Forward Counter
Add 3C header to the packet
Is initiated by last hop?
MAC Layer (Data Link)
Broadcast location discover packet
Location discover packet arrives
Broadcast location discover packet

Figure 4. Update the counters during the Location Discovery phase.

3.4 Counters Update during the Data Forwarding Phase

Once the location of the destination node is determined, the source node can start a communication session. In connectionless-oriented approach, nodes simply forward data packets without first establishing the link to the next node. Any node that happens to be in the general direction towards the destination node can compete for the “right” to forward data packets. When a source node *s* starts to send the data packet from routing layer to 3C Module, *s*’s 3C Module simply passes the data packet to the MAC layer without updating any counter.

3.4.1 Receive Data Packet

When a node *n* receives a data packet, its MAC Layer passes the data packet to its 3C Module. Then, *n*’s 3C Module updates the ***Forward Request Counter*** as follows:

1. 3C Module determines if the data packet corresponds to a communication session in *n*’s *Session Table*. If so, go to Step 2. Else, go to Step 3.
2. *n*’s 3C Module resets the *time to live* (*TTL*) *timer* of the corresponding entry in the *Session Table* and passes the data packet to the routing layer. Depend on different routing protocols, the data packet is either discarded or forwarded.
3. If the data packet is not belonged to any ongoing session in the *Session Table*, the 3C Module updates the *Session Table* and increases the ***Forward Request Counter*** by one. The 3C Module passes the Location Discovery packet to the routing layer for further processing (e.g., discard or forward data packet).

3.4.2 Forward Data Packet

Depend on the routing protocol, the data packet is either discarded or forward (see the Routing Layer in Figure 4). In connectionless-oriented approach, every node has equal probability of participate in the data forward procedure. If the routing layer decides to forward data packet, it returns a data packet to 3C Module. The 3C Module processes the data packet and updates the ***Forward Counter*** as follows:

1. 3C Module determines if the data packet matches any entry in the *Session Table*. If so, it increases the ***Forward Counter*** by one and passes the data packet to the MAC layer.
2. Else, the data packet is discarded. We discard any packets that are not in the *Session Table* due to the same reason as discussed in Section 3.3.3. A malicious node can generate dummy packets to avoid evoking the Misbehavior Detection procedure.

3.5 Initiate Misbehavior Detection

By modifying its own routing protocol, a malicious node can intentionally drop (i.e., discard) packets to save its power. However, in the connectionless-oriented approach, every node has an equal chance to participate in a forwarding process. Thus, 3C Module needs to determine to whether to “**invoke”** the **Misbehavior Detection procedure**. In order to determine if there is a need to invoke the Misbehavior

Detection procedure, 3C Module exams the 3C header in the Location Discovery packet and calculates two ratios, ***Forward Ratio*** (***FR***) and ***Request Ratio*** (***RR***) as follow:

- ***Forward Ratio*** $_i$ (***FR***$_i$) = $\dfrac{Forward\ Counter_i}{Forward\ Request\ Counter_i}$
- ***Request Ratio***$_i$ (***RR***$_i$) = $\dfrac{Location\ Discovery\ Counter_i}{Forward\ Counter_i}$

, where *i* is the node that initiated this Location Discovery packet (i.e., the source node).

When a node *n* receives a Location Discovery packet from a node *m*, *n*'s 3C Module checks if *m* is the initiator (i.e., source node) of this Location Discovery packet using the information included in the packet (see Section 3.3). If *m* is not the initiator, *n*'s 3C Module does not invoke the detection procedure. Then, this Location Discovery packet passes to the Counter Update procedure for further process (see Figure 4). If *m* is the initiator of this Location Discovery packet, *n*'s 3C Module checks the 3C header included in this Location Discovery packet for the following conditions:

1. $\mathbf{FR}_m < \mathbf{FR}_n$
2. $\mathbf{RR}_m > \mathbf{RR}_n$ * Initiate Detection Threshold

If one of the above condition is true, *n*'s 3C Module broadcasts a 3C packet (including *n*'s 3C information) to its one-hop neighbor nodes. When a node receives *n*'s 3C packet, it replies with its own 3C information. When *n* receives its neighboring nodes' replies, *n* calculates the ***Local Average Forward Ratio*** (***LAFR***). This ratio is calculated as follow:

LAFR$_n$ = $\dfrac{\sum_{i=1}^{k}(FR_i) + FR_n}{k+1}$, where *k* is number of neighboring nodes for *n* (excluding *m*).

In MANET, network conditions, such as density and congestion, can change dynamically. Thus, the ***Local Average Forward Ratio***$_n$ (***LAFR***$_n$) is merely the local network condition around *n*. If $\boldsymbol{FR_m} \geq \boldsymbol{LAFR_n}$, it means that network condition at area of *m* might be congested which causes *m* not forward packets. Thus, we do not need to invoke the Misbehavior Detection procedure. On the other hand, if $\boldsymbol{FR_m} < \boldsymbol{LAFR_n}$, then *m* might be misbehaving by not forwarding packets. In this case, *n* activates its **Detection Mode**. Notice that all the neighboring nodes of *m* and *n* can activate its **Detection Mode** (but not at same time) because their ***Forward Ratios*** are similar. When a node activates its **Detection Mode**, it continues to forward for other nodes except for the suspicious node.

To avoid evoking the Misbehavior Detection procedure, malicious nodes can initiate dummy packets to increase their own ***Forwarding Counter***. Although, by doing so, malicious nodes defeat the purpose of saving power. Nevertheless, 3C Module can prevent this misbehavior act by compare the outgoing packets against the *Session Table*. If the packet does not match any entry in the *Session Table*, 3C Module discards this dummy packet.

3.6 Detection Mode

The **Detection Mode** has two states: *Listening*-State and *Detecting*-State. Initially, a node in the Detection Mode is set to *Listening*-State. In the *Listening*-State, a node *n* waits for a random period of time. During this delay period of time, *n* does the following:

1. If *n* hears a Detection packet from another node to test node *m* (i.e., the suspect node), *n* resets the delay time. A Detection packet is generated by **Misbehavior Detection procedure** to test a suspicious node.
2. If *n* hears a Detection packet been forwarded by *m*, *n* exits the **Detection Mode**. By exiting the **Detection Mode**, *n* forwards *m*'s Location Discovery packet. Similarly, all other nodes that are in their Detection Mode (*Listen*-State) hear *m* forwarded the Detection packet will exist their Detection Mode.

At the end of delay period, node *n* enters the *Detecting*-State. In the *Detecting*-State, *n* invokes the **Misbehavior Detection procedure** to determine if *m* is a malicious node.

3.7 Misbehavior Detection Procedure

The detection mechanism can be implemented as a software application as proposed in [Buttyan, 00] for lower cost. Alternatively, it can also be implemented as a build-in component of the temper resistant module for better security. Without loss the generality, we base our discussion on the latter option.

The purpose of the **Misbehavior Detection procedure** is to detect uncooperative behaviors that result in disruption or degradation of data transmission. We focus on network layer attacks and do not address lower level threats such as physical layer jamming and MAC layer disruptions. The attacks contained by the Misbehavior Detection Module are as follows. First, the Misbehavior Detection procedure is invoked if there is a suspicion of dropping packets was detected during the location discovery phase. Second, the **Misbehavior Detection procedure** captures malicious users who deliberately discard packets that they are obligated to forward either for selfish purposes or to mount denial of service attacks.

When a node *n* invokes its **Misbehavior Detection procedure** to detect a suspect node *m*, the procedure is as follows:

For **CLA**:

1. n calculates a virtual link (see 1(a)) using the location information (i.e., cell ID) contained in m's Location Discovery packet.
2. Based on this *virtual link*, *n* generates a Detection packet (i.e., similar to regular data packet). The source location and destination location of this Detection packet are as follow:
   - Source node's location (*S_cell_ID*) of this Detection packet is the cell behind of *n*, relative to *m*.
   - Destination node's location (*D_cell_ID*) of this Detection packet is the cell behind of *m*, relative to *n*.
3. Next, *n* broadcasts this Detection packet. All the neighboring nodes of *m* are in Detection Mode and will not forward this Detection packet.
4. *n* waits for a *t* period of time (*t* = maximum delay time in the routing layer).

5. At the end of the delay, if *n* does not receive the Detection packet forwarded by *m* (i.e., *forward_ID* = *m*), *n* repeats the process again for two times (total of 3 times).

For **CBF**:

1. *n* calculates a *direction* (see 1(b)) to a destination which invented by *n* using the location information (i.e., the position of *m*) contained in *m*'s Location Discovery packet.
2. Based on the direction, *n* generates a Detection packet (i.e., similar to regular data packet). The source location and destination location of this Detection packet are as follow:
   - Source node's location (the position of Source node) of this Detection packet is the cell behind of *n*, relative to *m*.
   - Destination node's location (the position of Destination node) of this Detection packet is the cell behind of *m*, relative to *n*.
3. Next, *n* broadcasts this Detection packet. All the neighboring nodes of *m* are in Detection Mode and will not forward this Detection packet.
4. *n* waits for a *t* period of time (*t* = maximum delay time in the routing layer).
5. At the end of the delay, if *n* does not receive the Detection packet forwarded by *m* (i.e., *forward_ID* = *m*), *n* repeats the process again for two times (total of 3 times).

If *n* receives the detection packet which is forwarded by *m*, *n* (and all the neighboring nodes of *m*) exits the Detection Mode. *n* forwards *m*'s Location Discovery packet because *m* has passed *n*'s Misbehavior Detection procedure. If *n* does not receive the detection packet from *m*, *n* punishes *m* by discard *m*'s Location Discovery packet for period of $t_{punish} = C \times (\boldsymbol{LAFR_n} - \boldsymbol{FR_m})$. Thus, the punishment period is proportion to individual (misbehaving) node's misbehaved level.

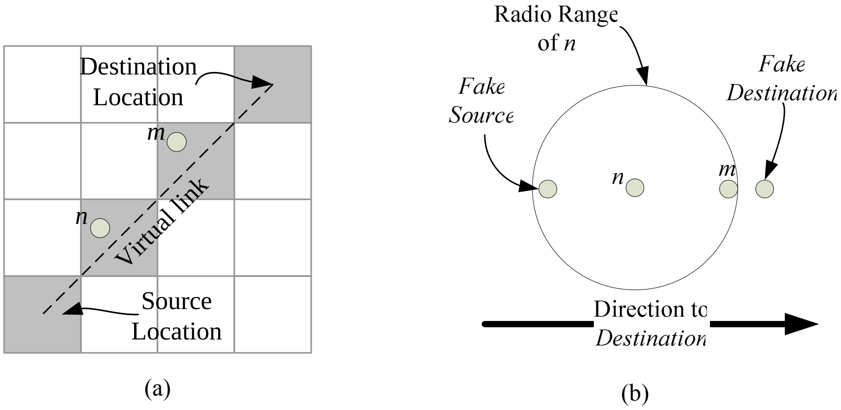


Figure 5. Virtual link for a Detection packet.

4 Experiment Results

We conducted various experiments to verify the effectiveness of the proposed 3CE (*3-Counter Enforcement*) scheme in enhancing performance of mobile ad hoc network. In this section, we first introduce the simulation setup and parameters. We then study the proposed technique based on various performance metrics.

4.1 Schemes Implemented

We implemented three schemes, namely the **reference** scheme, the **defenseless** scheme and the proposed **3CE** scheme, for performance evaluation. In the **reference** scheme, all the nodes act collaboratively and relay data for each other. In the **defenseless** scheme, a certain fraction of nodes are misbehaving as they failed to participate in forwarding procedure. In other words, these nodes discard any packets not destined at them. No detection or prevention mechanism is implemented so that the network is totally "defenseless". Finally, in the proposed **3CE** scheme, misbehaving nodes are detected and punished. A malicious node can recognize itself is been punished when Location Discovery packets of the node has been dropped four consecutively times. Once malicious nodes recognized themselves been punished, they participate in forwarding data to rejoin the network. We varied the *Initiate Detection Threshold* (*IDT*) from 1.0, 1.2, 1.4, and 1.6. This threshold determine percentage of a node require to participated in forward procedure in order not to initiate the 3C's detection procedure. For example, when the threshold set to 1.2, a node is allow of 20% of packet drop due to either network condition or mobility.

4.2 Simulation Setup

All the experiments were conducted using GloMoSim [Zeng, 98]. This simulator, developed at UCLA, is a packet-level simulator specifically designed for ad-hoc networks. It follows the OSI 7-layer network communication model. Although, popular simulators such as NS-2, OPNET Modeler, and GloMoSim provide advanced simulation environments to test and debug network protocols, we prefer GloMoSim due to its ability to handle high mobility of nodes and its scalability of handle large number of nodes and size of network area. Unlike other simulators, GloMoSim uses the parallel discrete-event simulation capability provide by Parsec [Bagrodia, 98].

Experiments were based on a mobile ad hoc network with 450 nodes and 90 communication sessions within a 1500 by 1500 meter two dimensional space. Each communication session, source node and destination node are randomly selected (i.e., both normal nodes and misbehaving nodes). Traffic applications are constant-bit-rate sessions. Each data packet is 512 bytes. For the CLA, each grid cell is 100 by 100 meter. The maximum delay time (*t*) is set to 2 *seconds*. All nodes employ 802.11 at the MAC layer. Each node has a radio range of about 250 meters. The random waypoint model was used to model the mobility of hosts. Multiple simulation runs (100 runs per setup on average) with different seed numbers were conducted for each scenario and collected data were averaged over those runs. The total simulation duration for each run was 60 minutes (3600 seconds). We varied the number of misbehaving nodes (i.e., 5%, 10%, 20%, and 30% of total number of nodes) and node mobility (i.e., 10 *m/s* to 25 *m/s* or 22 *mile/hr* to 56 *mile/hr*). Initially, misbehaving nodes drop all the received packets. Once misbehaving nodes been identified (i.e., all their Location Discovery packets are drop by other neighboring nodes), they behave normally until they are no longer identified as misbehaving nodes (i.e., their Location Discovery packets are forwarded by others).

4.3 Metric

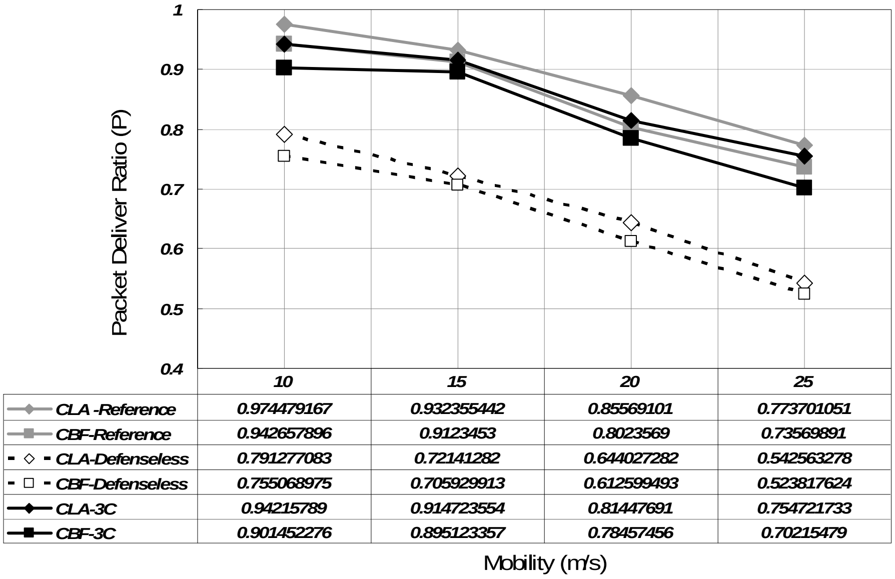
In the experiments, we evaluated the proposed scheme based on the following six metrics: (i) **Packet delivered ratio (*P*)**: The ratio of the data packets delivered to the destinations and the data packets generated by the CBR source. This measures the rate at which effective data transmission is performed. It is also a good indicator of the degree of collaboration among the nodes. (ii) **Misbehaving node detection ratio (*D*)**: The ratio of the number of misbehaving nodes that were correctly identified to the total number of misbehaving node that have actually acted uncooperatively during the simulation. (iii) **False accusation ratio (*F*)**: The ratio of the number of 3C Modules that incorrectly accused benign hosts to the overall number of misbehaving nodes that 3C Module identified. (iv) **Control overhead ratio (*C*)**: The ratio of the number of routing packets transmitted per distinct data packet delivered to a destination. (v) **End-to-end Delay (*E*)**: The number measured in *milliseconds*, includes detecting and processing malicious nodes delay, route discover latency, queuing delays, retransmission delay at the MAC, and propagation and transmission times. This measures the total delay time from a sender to a destination (without communication sessions that belong to misbehaving nodes). (vi) **Active Detection ratio (*A*)**: The ratio of the number of nodes activated their Detection Mode per misbehaving node's location discover packet.

4.4 Experimental Results

We present the simulation results in this section.

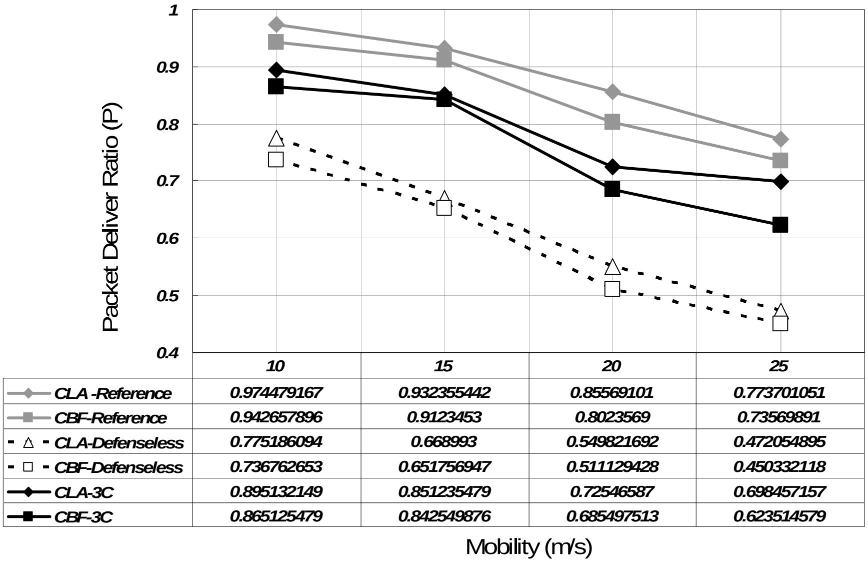
4.4.1 Packet Delivered Ratio

By employing the proposed scheme, significantly more data can be successfully delivered to the destinations since nodes are now required to participating in data forwarding. Figure 6 and Figure 7 depict the practical scenarios where the number of malicious node is 10% and 20% of the total nodes. We observe in the case of fewer malicious nodes (less than 10%), the two protocols with 3CE (i.e., *CLA-3C* and *CBF-3C* with the *Initiate Detection Threshold* = 1.2) have very close throughput to the references CLA and CBF (i.e., *CLA-Reference* and *CBF-Reference*). Notice that the performance of 3CE scheme is slightly less than the reference scheme. This is due to two reasons: 1) misbehavior nodes are not 100% detected (i.e., see section 4.4.2, the 3CE's misbehaving detection ratio is about 87%); and 2) the false accusation ratio is not 0% (i.e., see section 4.4.3, some nodes are been miss identified as malicious nodes). Also, notice that in the reference scheme, even all the nodes act collaboratively and relay data for each other, network condition (e.g., channel error, congestion, and mobility) are still the main causes for packet loss. Never the less, the results show that the proposed 3CE scheme can minimize the effect of malicious nodes to the network. In both cases, the proposed technique improves the deliver ratio by more than 25% compare to the defenseless scheme.



	10	15	20	25
CLA -Reference	0.974479167	0.932355442	0.85569101	0.773701051
CBF-Reference	0.942657896	0.9123453	0.8023569	0.73569891
CLA-Defenseless	0.791277083	0.72141282	0.644027282	0.542563278
CBF-Defenseless	0.755068975	0.705929913	0.612599493	0.523817624
CLA-3C	0.94215789	0.914723554	0.81447691	0.754721733
CBF-3C	0.901452276	0.895123357	0.78457456	0.70215479

Figure 6. Packet Deliver Ratio (P) with 10% Malicious Nodes.

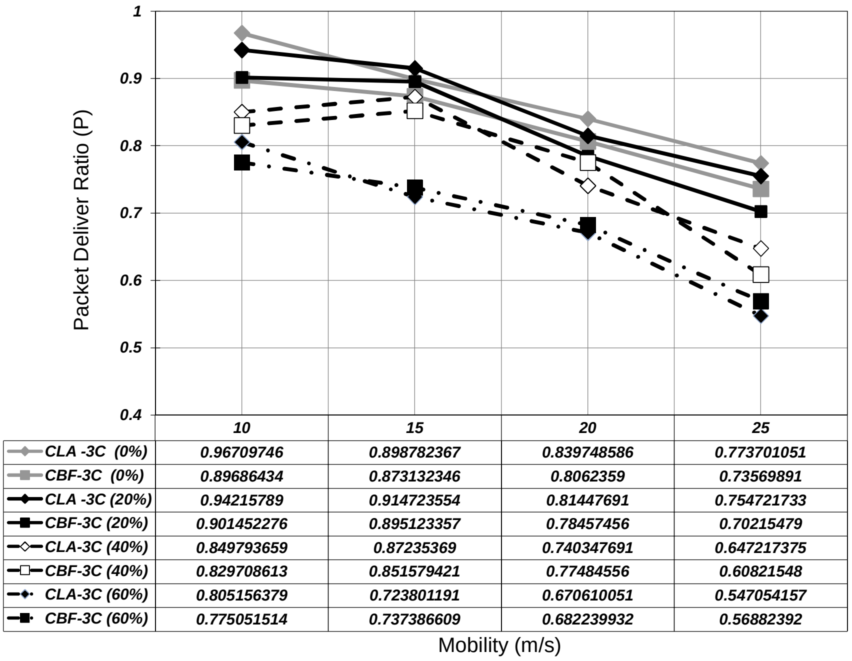


	10	15	20	25
CLA -Reference	0.974479167	0.932355442	0.85569101	0.773701051
CBF-Reference	0.942657896	0.9123453	0.8023569	0.73569891
CLA-Defenseless	0.775186094	0.668993	0.549821692	0.472054895
CBF-Defenseless	0.736762653	0.651756947	0.511129428	0.450332118
CLA-3C	0.895132149	0.851235479	0.72546587	0.698457157
CBF-3C	0.865125479	0.842549876	0.685497513	0.623514579

Figure 7. Packet Deliver Ratio (P) with 20% Malicious Nodes.

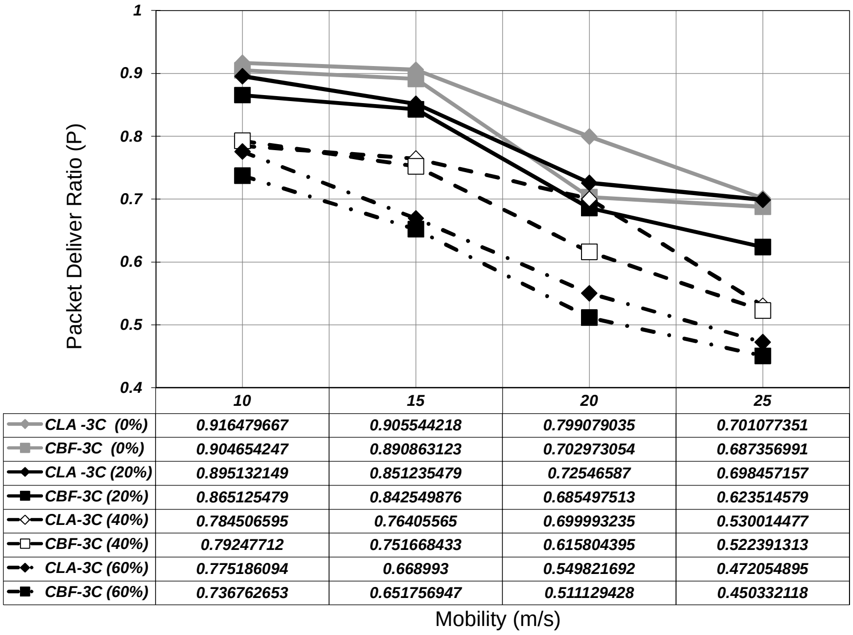
In Figure 8 and Figure 9, we varied the *Initiate Detection Threshold* from 1.0, 1.2, 1.4, and 1.6. This mean a node is allow to drop from 0%, 20%, 40%, and 60% of total packets received without initiate 3CE's detection mode. We observe in the case of *Initiate Detection Threshold* = 1.6 (60% of tolerable drop rate), the performances of the proposed technique decreased to the defenseless scheme. However, if we decrease the *Initiate Detection Threshold* = 1.0 (0% of tolerable drop rate), we do not gained additional improvement on the deliver ratio. In fact by doing so, the performances of Misbehaving node detection ratio (*D*), False accusation ratio (*F*), Control overhead ratio (*C*), and Active Detection ratio (*A*) are decreased (see following sections). Based on the simulation results, the ideal *Initiate Detection Threshold* is 1.2 (20% of tolerable drop rate) with probability of error of 5%.

Another important factor to the performance of packet deliver ratio is the speed of mobility. Due to mobility of mobile hosts, addressing frequent and unpredictable topology changes is fundamental to MANET research. As the mobility of node (e.g., speed) increase, the performance of all three schemes (i.e., 3CE, reference, and defenseless) are decreased. Similarly when we increased mobility and number of malicious nodes (see Figure 8 and Figure 9), the packet deliver ratio is also decreased as the result. However, consider of mobility increased from 10 *m/s* (or 22 *miles/hour*) to 25 *m/s* (or 56 *miles/hour*), the deliver ratio is only drop average 20%. Thus, the protocol is still suited for many applications (e.g., video and audio) with error correction code.



	10	15	20	25
CLA -3C (0%)	0.96709746	0.898782367	0.839748586	0.773701051
CBF-3C (0%)	0.89686434	0.873132346	0.8062359	0.73569891
CLA -3C (20%)	0.94215789	0.914723554	0.81447691	0.754721733
CBF-3C (20%)	0.901452276	0.895123357	0.78457456	0.70215479
CLA-3C (40%)	0.849793659	0.87235369	0.740347691	0.647217375
CBF-3C (40%)	0.829708613	0.851579421	0.77484556	0.60821548
CLA-3C (60%)	0.805156379	0.723801191	0.670610051	0.547054157
CBF-3C (60%)	0.775051514	0.737386609	0.682239932	0.56882392

Figure 8. Packet Deliver Ratio (P) with 10% Malicious Nodes.



	10	15	20	25
CLA -3C (0%)	0.916479667	0.905544218	0.799079035	0.701077351
CBF-3C (0%)	0.904654247	0.890863123	0.702973054	0.687356991
CLA -3C (20%)	0.895132149	0.851235479	0.72546587	0.698457157
CBF-3C (20%)	0.865125479	0.842549876	0.685497513	0.623514579
CLA-3C (40%)	0.784506595	0.76405565	0.699993235	0.530014477
CBF-3C (40%)	0.79247712	0.751668433	0.615804395	0.522391313
CLA-3C (60%)	0.775186094	0.668993	0.549821692	0.472054895
CBF-3C (60%)	0.736762653	0.651756947	0.511129428	0.450332118

Figure 9. Packet Deliver Ratio (P) with 20% Malicious Nodes.

4.4.2 Misbehaving Node Detection Ratio

We list the results of misbehaving node detection ratio for various simulation scenarios in Table 1. They indicate that the proposed misbehaving node detection mechanism is very effective. In most cases with the *Initiate Detection Threshold* = 1.2 (or 20% of tolerable drop rate), the 3CE's detection ratio is about 87%. However, when the threshold increased to 1.6 (60% of tolerable drop rate), the 3CE's detection ratio decrease to about 50%. This indicated that it important to select the acceptable threshold for the proposed technique. The results (with correct threshold selected) demonstrate that on-demand misbehaving node detection is applicable. Since the proposed 3CE technique can adapt by the connectionless oriented approach, it is ideal for highly dynamic MANETs such as vehicle-to-vehicle networks.

Speed (m/s)	10								15							
Tolerabe Drop rate (%)	0%		20%		40%		60%		0%		20%		40%		60%	
Protocol	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF
5% misbehaving nodes	96%	96%	89%	87%	77%	75%	59%	62%	96%	94%	88%	88%	75%	76%	55%	54%
10% misbehaving nodes	93%	94%	93%	88%	77%	74%	60%	61%	92%	93%	91%	89%	77%	74%	54%	52%
20% misbehaving nodes	94%	93%	91%	90%	75%	72%	58%	60%	93%	92%	85%	87%	73%	73%	54%	52%
30% misbehaving nodes	93%	91%	91%	88%	68%	66%	52%	54%	91%	92%	87%	85%	66%	65%	50%	49%
Speed (m/s)	20								25							
Tolerabe Drop rate (%)	0%		20%		40%		60%		0%		20%		40%		60%	
Protocol	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF
5% misbehaving nodes	90%	88%	83%	81%	71%	68%	52%	55%	84%	82%	81%	80%	68%	66%	52%	53%
10% misbehaving nodes	92%	91%	86%	86%	71%	70%	52%	49%	85%	87%	88%	85%	65%	63%	60%	59%
20% misbehaving nodes	89%	90%	89%	88%	68%	66%	48%	47%	87%	85%	87%	86%	58%	62%	54%	55%
30% misbehaving nodes	88%	85%	84%	82%	59%	62%	45%	43%	86%	84%	85%	80%	50%	48%	34%	32%

Table 1: Detection Ratio (D) of CLA and CBF with 3CE.

4.4.3 False Accusation Ratio

We report the false accusation ratios of the proposed 3CE scheme under various scenarios in Table 2. We conclude that in all node mobility scenarios with the *Initiate Detection Threshold* = 1.2 (or 20% of tolerable drop rate) the false accusation ratio is very low. We observe that this ratio is higher when the speed of nodes is increased. This is due to the fact that some of the suspect nodes moved out of the detection node's radio range and were thus incorrectly classified by 3CE's Misbehaving Detection procedure as misbehaving nodes, thereby lifting the false accusation ratio. In addition, by decrease the *Initiate Detection Threshold*, the false accusation ratio is increased. The reason is that nodes drop packets due to the network condition were identify as misbehaving nodes. Nevertheless, further investigation of simulation log files shows that under simulation configuration with the *Initiate Detection Threshold* = 1.2, on average less than four nodes was incorrectly accused. Both results indicate that the proposed detection mechanism is able to detect most of the in-cooperative nodes with very low false accusation ratio.

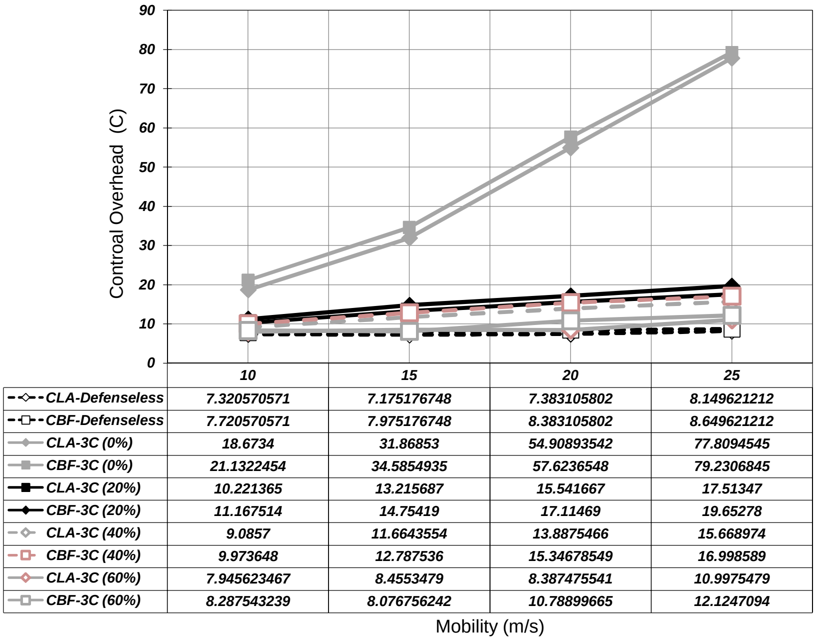
Speed (m/s)	10								15							
Tolerabe Drop rate (%)	0%		20%		40%		60%		0%		20%		40%		60%	
Protocol	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF
5% misbehaving nodes	8%	7%	0%	1%	3%	5%	8%	7%	5%	6%	2%	2%	5%	5%	5%	6%
10% misbehaving nodes	8%	11%	1%	1%	4%	7%	8%	7%	7%	6%	2%	2%	5%	6%	7%	6%
20% misbehaving nodes	9%	10%	1%	1%	5%	7%	8%	8%	9%	11%	1%	1%	6%	7%	9%	8%
30% misbehaving nodes	12%	11%	2%	3%	6%	7%	8%	9%	17%	16%	2%	2%	10%	11%	11%	11%

Speed (m/s)	20								25							
Tolerabe Drop rate (%)	0%		20%		40%		60%		0%		20%		40%		60%	
Protocol	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF	CLA	CBF
5% misbehaving nodes	7%	8%	3%	2%	6%	6%	7%	8%	12%	11%	2%	1%	7%	9%	9%	8%
10% misbehaving nodes	8%	8%	2%	2%	7%	6%	8%	8%	14%	16%	3%	3%	8%	9%	9%	10%
20% misbehaving nodes	10%	11%	2%	2%	7%	8%	10%	11%	18%	17%	2%	2%	11%	10%	12%	13%
30% misbehaving nodes	13%	16%	4%	5%	11%	9%	10%	11%	23%	26%	5%	5%	11%	11%	13%	13%

Table 2: False Accusation Ratio (F) of CLA and CBF with 3CE.

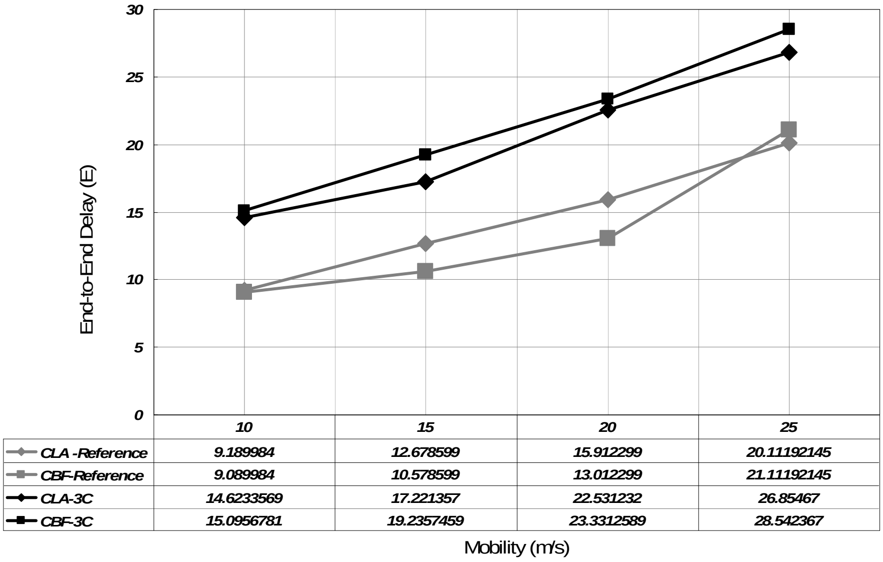
4.4.4 Control Overhead Ratio

With 20% of malicious nodes and the *Initiate Detection Threshold* = 1.2 (or 20% of tolerable drop rate), we observe that the Control Overhead Ratio is higher when the speed of nodes is increased (see Figure 10). Similar to False Accusation Ratio, this is due to the fact that some of the suspect nodes moved out of the detection node's radio range and were thus cause some nodes to invoke 3CE's Misbehaving Detection procedure, thereby lifting the Control Overhead Ratio. However, this is inevitable in most on-demand misbehaving node detection approaches. With the *Initiate Detection Threshold* = 1.0 (or 0% of tolerable drop rate), we also observe that the Control Overhead Ratio is increased four times higher compare to the simulation results with the *Initiate Detection Threshold* ≥ 1.2. Again similar to False Accusation Ratio, more incidences of Detection procedure were invoked to due to the low threshold.



	10	15	20	25
CLA-Defenseless	7.320570571	7.175176748	7.383105802	8.149621212
CBF-Defenseless	7.720570571	7.975176748	8.383105802	8.649621212
CLA-3C (0%)	18.6734	31.86853	54.90893542	77.8094545
CBF-3C (0%)	21.1322454	34.5854935	57.6236548	79.2306845
CLA-3C (20%)	10.221365	13.215687	15.541667	17.51347
CBF-3C (20%)	11.167514	14.75419	17.11469	19.65278
CLA-3C (40%)	9.0857	11.6643554	13.8875466	15.668974
CBF-3C (40%)	9.973648	12.787536	15.34678549	16.998589
CLA-3C (60%)	7.945623467	8.4553479	8.387475541	10.9975479
CBF-3C (60%)	8.287543239	8.076756242	10.78899665	12.1247094

Figure 10. Control Overhead Ratio (C) with 20% Malicious Nodes.



	10	15	20	25
CLA -Reference	9.189984	12.678599	15.912299	20.11192145
CBF-Reference	9.089984	10.578599	13.012299	21.11192145
CLA-3C	14.6233569	17.221357	22.531232	26.85467
CBF-3C	15.0956781	19.2357459	23.3312589	28.542367

Figure 11. End to End Delay (E) with 20% Malicious Nodes.

4.4.5 End-to-End Delay

We report the increasing of end-to-end delay in Figure 11. With 20% of malicious nodes, we observe that the proposed scheme incurs minimum end-to-end delay. In most of cases, the length of delay increases approximately five *milliseconds* compared the reference schemes. This can due to the fact that other nodes can continue to forward data packet while one node is detecting a malicious node. Also, malicious nodes are unable to utilize the network resource once they are identified. Since we punish the misbehaving nodes by not forwarding their Location Discovery packet for a period of time, we did not include the communication sessions which the source nodes are misbehaving nodes. In addition, the *Initiate Detection Threshold* does not affect end-to-end delay due the characteristics of the connectionless-oriented approach where anyone within the virtual router (i.e., CLA) or the neighboring node within the general direction of destination (i.e., CBF) can alternate in forwarding data toward the next virtual router or the next node.

4.4.6 Active Detection Ratio

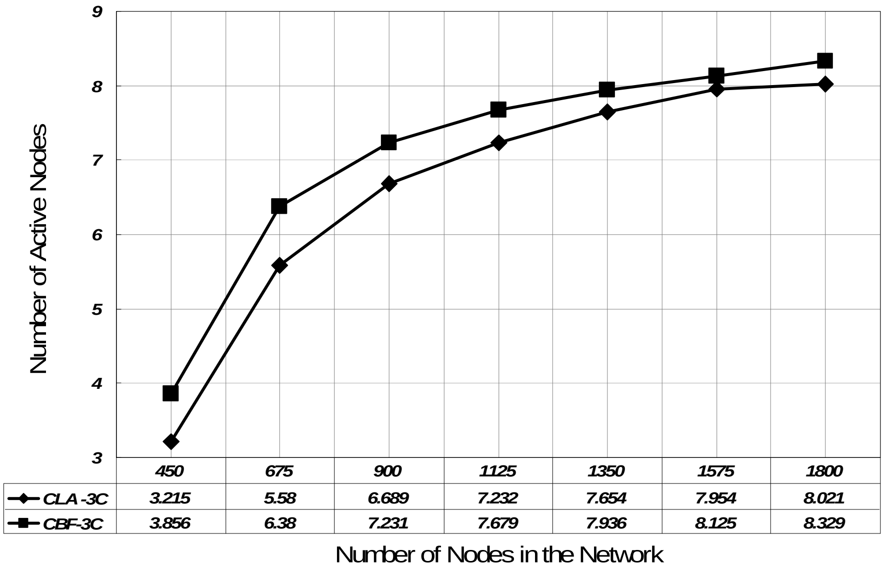
With speed of 20 *m/s* and 20% of malicious nodes, we observe that the number of nodes activated Detection Mode per malicious node's location discover packet (that attempt to establish a connection) becomes fixed even the number of nodes in the network increased from 450 nodes to 1800 nodes (see Figure 12). In fact, if a malicious node is stationary, the maximum number of neighboring nodes that are in the Detection Mode (i.e., *Detecting*-State) is six (see Figure 13 (a)). If a malicious node is moving at speed of 20 *m/s*, then the moving rang (i.e., a circle with radius of *r*) within the maximum delay time (*t* = 2 *seconds*) of the Detection Mode is as follow:

$$r = speed * time = 20(m/s) * 2(s) = 40(m)$$

With radio range of a node is 250 meters; the radius of circular area of the maximum area of neighboring nodes that can activate Detection Mode is as follow:

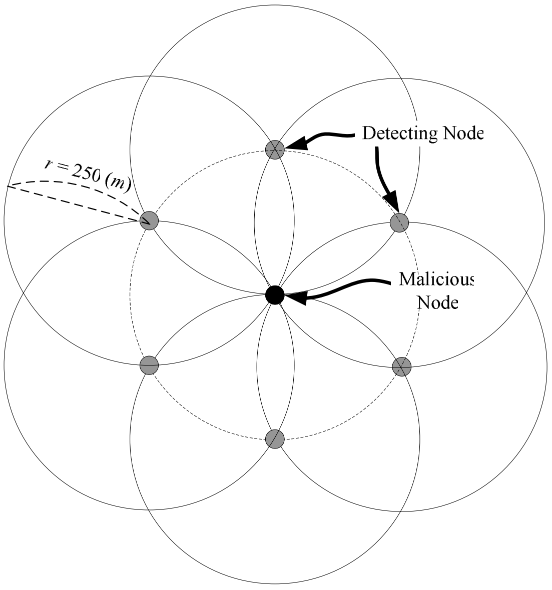
$$r_{Detection} = r + \text{radio range } = 40(m) + 250(m) = 290(m)$$

Thus, the maximum number of neighboring nodes that are in the Detection Mode is seven nodes (see Figure 13 (b)). In order for a malicious node to move out of area where its neighboring nodes have activated the Detection Mode, the malicious node needs to travel of 540 *meters* (i.e., 290 *m* + 250 *m*). With maximum moving speed of 20 m/s, the time a malicious node to move out of this area is 27 seconds (i.e., 540(*m*) / 20 (*m/s*)). Thus, the upper bond of Active Detection Ratio (*A*) is 7 nodes per 27 seconds (or 0.26 nodes per second). This confirms with our simulation study. In fact, the result in Figure 12 shows that our approach is able to adapt under high mobility (i.e., variety of applications – vehicular networks) and high density networks (i.e., scalable).

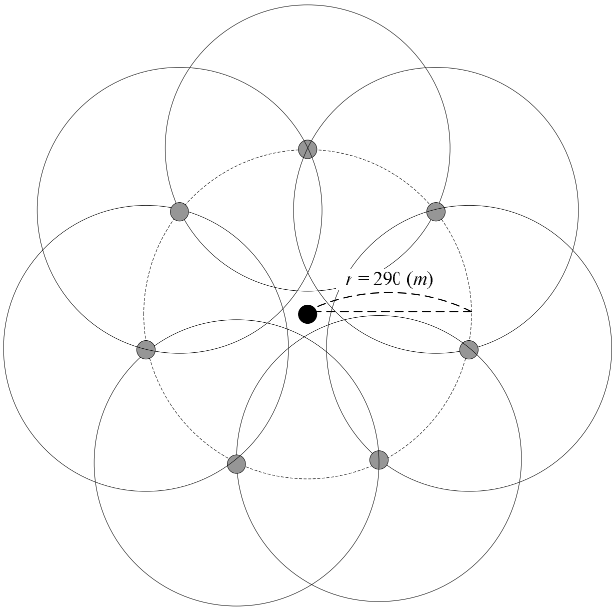


	450	675	900	1125	1350	1575	1800
CLA -3C	3.215	5.58	6.689	7.232	7.654	7.954	8.021
CBF-3C	3.856	6.38	7.231	7.679	7.936	8.125	8.329

Figure 12. Active Detection Ratio (A) with 20 m/s and 20% malicious nodes.



(a)



(b)

Figure 13. Number of detecting nodes needed per malicious node at different speed.

5 Conclusion

In this paper, we proposed an efficient 3CE (*3-Counter Enforcement*) scheme to enforce collaboration for the connectionless-oriented approach (i.e., CLA and CBF) in mobile ad hoc network. Our contributions are as follows. 1) We introduce an on-demand approach to misbehaving-node detection for the connectionless-oriented approach. Since the connectionless-oriented approach addresses highly dynamic networks (i.e., vehicle-to-vehicle networks), the existing misbehaving-node detection techniques are not suitable. Our approach supports this type of routing protocol under high mobility environments. 2) Each node maintains three counters to represent its own status (i.e., reputation). Since nodes only determine their neighboring nodes' counters information when a location discovery phase, no additional information is needed under a normal operation (i.e., nodes behave normally). 3) With large number of nodes and high mobility, the proposed approach enforces the cooperation on-demand with minimum increase of delay.

We conducted various experiments to study the effectiveness and efficiency of the proposed 3CE technique. The simulation results indicated that the proposed technique is very effective in enforcing collaboration. The degree of collaboration is significantly strengthened as the network throughput is greatly improved compare to a defenseless network. Such improvement is accomplished with almost no false accusation of cooperative nodes. As of efficiency, the proposed scheme incurs minimum delay.

References

[Bagrodia, 98] Bagrodia, R., Meyer, R., Takai, M., Chen, Y. A., Zeng, X., Martin J., Song, H. Y. Parsec: A Parallel Simulation Environment for Complex Systems. IEEE Computer. Vol 31, Issue 10, pp. 77 – 85. Oct 1998.

[Blazevic, 05] Blazevic, L., Le Boudec, J., and Giordano, S. A Location-Based Routing Method for Mobile Ad Hoc Networks. Transactions on Mobile Computing. Vol. 4, No. 1.pp. 97-110, March 2005.

[Buchegger, 02] Buchegger, S. and Boudec, J. L. Performance Analysis of the CONFIDANT Protocol: Cooperation of Nodes – Fairness in Dynamic Ad Hoc Networks. In Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, CH, June 2002.

[Buttyan, 00] Buttyan, L. and Hubaux, J. Enforcing Service Availability in Mobile Ad Hoc WANs. In Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC), Boston, MA, USA, August 2000.

[Fubler, 03] Fubler, H., Widmer, J., Kasemann, M., Mauve, M., and Hartenstein, H. Contention-Based Forwarding for Mobile Ad-Hoc Networks, Elsevier's Ad-Hoc Networks, Vol 1, no 4, pp. 351-369, 2003.

[Ho, 04] Ho, Y.H., Ho, A.H., Hua, K.A., and Hamza-Lup, G.L. A Connectionless Approach to Mobile Ad hoc Networks. Proc. of Ninth International Symposium on Computers and Communications (ISCC), Vol 1, pp. 188-195, Alexandria, Egypt, 2004.

[Ho, 07] Ho, Y.H., Ho, A.H., and Hua, K.A. Connectionless Protocol – A Localized Scheme to Ad Hoc Network. Proc. of International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC) Vol. 2, Issue 1/2, pp. 21 – 35, 2007.

[Jiang, 05] Jiang, N., Sheu, S., Hua, K. A., and Ozyer O. A Finite-State Model Scheme for Efficient Cooperation Enforcement in Mobile Ad Hoc Networks. In Proceedings 11th International Conference on Parallel and Distributed System, Fukuoka, Japan, 2005.

[Karp, 00] Karp, B. and Kung, H. T. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In Proc. of MOBICOM '00, page 243-254, Boston, MA, U.S.A., August 2000.

[Ko, 98] Ko, Y. B. and Vaidya, N. H. Location-Aided Routing (LAR) in Mobile Ad Hoc Network, Proc. of ACM/IEEE MOBICOM '98, pp. 66-75, Dallas, Texas, United States, October 1998.

[Marti, 00] Marti, S., Giuli, T.J., Lai, K., and Baker, M. Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In Proceedings of MOBICOM 2000, page 255-265, 2000.

[Michiardi, 02] Michiardi, P. and Molva, R.. CORE: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Network. 6th IFIP Conference on Security Communications and Multimedia (CMS 2002), Portoroz, Slovenia, 2002.

[Niculescu, 03] Niculescu, D. and Nath, B. Trajectory Based Forwarding and Its Applications, Proc. of ACM/IEEE MOBICOM '03, pp. 260-272, San Diego, California, September 2003.

[Pfitzmann, 97] Pfitzmann, A., Pfitzmann, B., Schunter, M., and Waidner, M. Trusting Mobile User Device and Security Modules. In Computer, pp. 61-68. IEEE, February 1997.

[Smith 97] Smith, B. R., Murthy, S., and Garcia-Luna-Aceves, J. J. Securing Distance-Vector Routing Protocols. In Proc. of Internet Society Symposium on Network and Distributed System, San Diego, CA, pp. 85-92, Feb. 1997.

[Yi, 01] Yi, S., Naldurg, P., and Kravets, R. Security-Aware Ad-Hoc Routing for Wireless Networks. MobiHoc Poster Session, 2001.

[Zeng, 98] Zeng, X., Bagrodia, R., and Gerla, M. "GloMoSim: a library for parallel simulation of large-scale wireless network," *Proceedings of the twelfth workshop on Parallel and distributed simulation*, pp. 154-161, May 1998, Banff, Alberta, Canada.

[Zhou, 99] Zhou, L., and Haas, Z.. Securing Ad Hoc Network. In IEEE Network magazine, special issue on network security, Vol. 13, No. 6 November/December, pp. 24-30, 1999.