Trust Establishment in Ad Hoc Networks by Certificate Distribution and Postponed Verification

Dawoud S.Dawoud[#1], Ashraf Sulaiman[#2], Richard Gordon [#3],

[1]Professor, Dean of Faculty of Applied Science, National University of Rwanda, Butare, Rwanda
Corresponding author email: dsdawoud@gmail.com
[2]Senior lecturer, Faculty of Applied Science, NUR, Rwanda
[3]Eng. (MSc.) South Africa

ABSTRACT
Trust establishment in wireless ad hoc networks is a challenge because of its unique characteristics. These include the lack of a central authority and the autonomous, dynamic nature of these networks. Such characteristics of ad hoc networks result in poor connectivity and routing failure. Security of the network can be provided by a certificate based model but the key management stays as a difficulty in wireless ad hoc networks. A key management scheme is proposed which realizes certificate distribution and verification. The proposed key management scheme is an on-demand and fully distributive which is suitable for the wireless ad hoc network environment. It establishes trust on the routing layer exclusively. Trust and route establishment are achieved simultaneously with reduced dependency between the security and routing mechanisms. Distribution and verification of keying material places delays upon the delivery of secure communication routes. Simulations show the overhead of the proposed scheme and that it has negligible impact on network performance while providing trust establishment for the network.

Keywords: Ad hoc networks; direct and indirect trust; fully distributive key management; on-demand secure-trust; trust establishment

1.0 INTRODUCTION

Wireless ad hoc networks are complex networks which have little or no existing network infrastructure. This lack of fixed network architecture creates complex security problems. Building trust between the nodes forming the network is a major issue in ad hoc security. The term secure trust is defined as the "belief by a trustor with respects to the competence, honesty, security and dependability of a trustee within a specific context."(Grandison,2003). There are two main approaches for trust establishment: certificate based trust models (Capkun *et al*, 2003) (Zhou & Haas, 1999) and conduct based trust models (Tanabe and Aida, 2007)(Theodorakopoulos and Baras, 2006). Certificate based trust models use vital keying material to provide trust. There are two trust variables: direct trust and indirect trust. Direct trust is a result of independent or local trust evaluation, between two immediate nodes. Indirect trust is evaluated using the advice from other nodes. In the context of certificate based trust, direct trust is defined as trust between local neighbours and indirect trust is created by certificate chaining. Key management is central to certificate based trust establishment (Capkun *et al*, 2003) (Zhou and Haas, 1999)(Capkun *et al*, 2006). One primary task of key management is the distribution of the keying material, e.g. self-certificates. In a fixed network an on-line trusted authority is present to perform key management tasks. This is not possible in an ad hoc network which lacks a central trusted authority or fixed network infrastructure. Achieving key management in mobile ad hoc networks is a challenge due to the lack of a central authority and the autonomous, dynamic nature of these networks which result in poor connectivity and routing failure. Many secure routing protocols for mobile ad hoc networks are published, e.g. *SAODV* (Perkins *et al,* 2003), *SEAD* (Hu *et al*, 2002), *ARIADNE* (Hu *et al*, 2005), and *endairA* (Acs *et al*, 2006). Most of these assume pre-existing and pre-sharing keying relationships. Key management proposed in (Awerbuch *et al*, 2008) (Zapata, 2006) operates on the routing layer to achieve key distribution. The required certificates are appended to all routing request in an effort to distribute keying material during

the route establishment phase. This approach is not ideal for an on-demand ad hoc network environment because it results in flooding the network with route request during its route discovery phase. In general, we can identify two main approaches to solve the key management problem in ad hoc networks. The first approach is the partial distributive scheme, for example, the schemes proposed in (Zhou and Haas, 1999)(Awerbuch *et al*, 2008). This approach distributes the functionality of the trusted network authority amongst a limited number of nodes. The second approach is a fully distributive scheme (Capkun *et al*, 2003) (Zapata, 2006) which distributing the security responsibilities across the entire network. In a fully distributive scheme each node is considered to be the centre of its own world, and is responsible for its own secure communication (Capkun *et al*, 2003) . The scheme proposed by Capkun (Capkun *et al*, 2006) does not consider the delay incurred from the key management task of verification, assuming it to be negligible. Existing models have such delayed bootstrapping security phases that security is only delivered after an initial time of setup. This creates a window period of weaken security or a window period of restricted communication (Tanabe and Aida, 2007)(Capkun *et al*, 2006). The aim of our paper is to design a key management scheme that can be used to distribute and verify certificates in a wireless on-demand ad hoc network, with negligible affects upon routing performance. The proposed scheme establishes trust by distribute and verify certificates for all the nodes in a network with the following constraints:

- The key management scheme is to operate in an on-demand environment, exclusively on the routing layer.
- The key management scheme is to distribute and verify certificates between local and remote nodes providing direct and indirect trust relationships respectively.
- Each node in an indirect trust chain must verify its neighbour, the originator and destination node's certificates, before the trust chain is secure.
- The key management scheme aims to minimize the security overheads which affect the network routing performance. These overheads include certificate verification and distribution delays.
- The key management scheme should avoid altering the routing mechanism, and strive for independence between routing and trust establishment. Routing packet size is not to be extended to incorporate security information.
- The certificate scheme is to be designed in a fully distributive manner with no existence of an on-line trust authority or prior trust relationships.
- Security should be available as a node enters a network with a seemingly timeless bootstrapping phase for security.
- The key management scheme should be robust to poor connectivity and routing failure due to shifting mobility, error-prone wireless channels and traffic congestion which are natural characteristics of wireless ad hoc networks.

The proposed scheme is called Direct, Indirect Trust Distribution (DITD) and it follows the procedure outlined in Section-2. The paper is structured in the following manner: In Section-2 the Direct, Indirect Distribution scheme is proposed, describing the distribution and post verification mechanisms. Section-3 includes the implementation, simulation and evaluation of DITD's performance. Section-4 provides the closing conclusions.

PROPOSED SCHEME

2.1System Model

To fulfil the constraints given in Section I, we assume in this section a system model in which there is no pre-existing infrastructure and no online trusted third party present during communication. The model is a fully distributive network of wireless nodes using an ad hoc on-demand routing mechanism. The model assumes that nodes

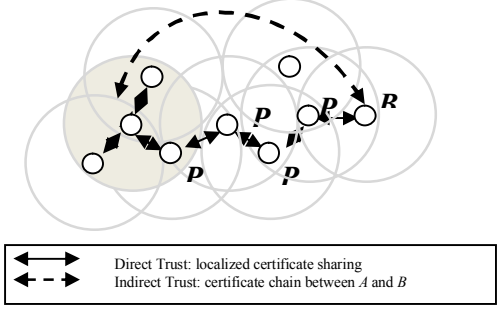

Fig-1 Direct and Indirect trust establishment

have their own keying material before joining the network generated by a fully self organized mobile ad hoc network (Capkun *et al*, 2003) , or by an off-line authority issuing keying material before a node enters the network (Capkun *et al*, 2006). Each node is assumed to have a public and private key pair, a certificate binding the public key and user identification of the node, and a set of network security parameters common to all nodes in the network. Secure communication is requested from the start to the end of the network lifetime, unlike (Tanabe and Aida, 2007)(Capkun *et al*, 2006) which is flawed by its initial setup phase with weak security.

2.2 Proposed DITD Model

The proposed Direct, Indirect Trust Distribution Model (DITD) aims to distribute and verify self certificates to create direct and indirect trust relationships between nodes. DITD is a certificate based trust model which works with existing mobile ad hoc routing schemes. It is not specific for a single routing protocol but its principals can be applied to any routing scheme. In the following we introduce the proposed scheme in AODV environment.

AODV (Perkins *et al*, 2003) routing procedure has three stages: sending the request message; receiving the request message; and sending the reply message. In the first stage, the originator node *A* requests communication with destination node *B* by broadcasting a routing request *RREQ* into the network. This request is forwarded by intermediate nodes and propagated through the network to *B*. When the *RREQ* message is received by an intermediate node *P*, it may have been sent by *A* or forwarded by a neighbouring node *NP*. Upon receiving the *RREQ* message stage two begins. At stage two a reverse route to *A* is then set up and *P* checks if it is the destination *B* or has a fresh route to the destination node *B*. If not, then the *RREQ* is further broadcast by *P* and propagates until the destination is found. When the destination or a fresh route to the destination is found, stage three commences. A reply message *RREP* is propagated along the reverse route until it reaches the originator node *A* establishing the communication route. When a node receives a routing control packet, and before that packet is processed, DITD sends certificate requests using separate unicast messages. The self certificate distribution is added at stage two and stage three; the receiving of the route request and the sending of the reply message stages. At stage two, upon receiving a route request packet, before this packet is processed and the routing table updated, direct trust and indirect trust establishment is set up. The proposed scheme is subdivided into three parts: Direct trust establishment, indirect trust establishment and the post verification optimization.

2.2.1 Direct Trust

At stage two, direct trust relationships are made by allowing the neighbouring nodes to exchange certificates. When intermediate node *P* receives a route request *RREQ* it first checks its certificate repository for the certificate of the neighbour, *NP*, who forwarded the request. If it does not possess such a certificate, $Cert_{NP}$, a local self certificate exchange is done between node *P* and its neighbour *NP* using two unicast messages. The exchange of certificates follows the *RREQ*. This will flood the network in search of a route to the destination node. Direct trust establishment is illustrated in Figure 1. What can be expected is an increased initial packet overhead.

2.2.2 Indirect Trust

Similar to direct trust establishment, at stage two, node *P* searches for the originator's certificate, $Cert_A$. If it is not found, node *P* sends a unicast certificate request for $Cert_A$ to *NP* whose address can be found at the next hop on the reverse route. This propagates $Cert_A$ to the destination *B*. For indirect certificate trust to be established originator *A* is required to possess the destination's certificate, $Cert_B$, as well. By not appending the certificate to the route requests dependency is reduced between the route establishment and trust establishment.

At stage three, sending the reply message, the indirect trust establishment is completed. Sending a reply is guided by two conditions. First when the destination node is found and secondly when a fresh route to the destination node is found. For the first condition, the reverse route to *A* is already setup with localized direct trust existing between nodes on the route; therefore a trusted certificate chain of nodes is available towards the originator node *A*. It is required only that the certificate of the destination node, $Cert_B$, to be piggy backed on the routing reply message *RREP* toward *B*. Each intermediate node stores $Cert_B$ and updates its certificate repository. For the second condition, if a fresh route to *B* is found, there exists a route from intermediate node *P* to destination *B* and a route from *P* to *A*. Both routes have localized direct trust existing already, so the two routes can be view as certificate chains. Two *RREP* messages are then propagated, one toward *B* with the $Cert_A$ appended and one toward *A* with the $Cert_B$ appended. Indirect trust is therefore set up by certificate chaining as illustrated in Figure 1.

2.2.3 Verification

Verification upon an indirect trust chain requires each node to verify its chain neighbour, the originator and destination node's certificate. Ideally verification will take place immediately after a certificate is received but the processing of a single verification results in a computational delay. For application specific networks that are time dependent like military automation networks, a delay of even milliseconds is critical. DITD provides optimized verification by allowing routing messages to be forwarded pending verification confirmation.

Verification for direct trust establishment can be done immediately without incurring a delay upon the routing mechanism. This is because the localised certificate messages are separate and independent from the request messages. Furthermore during route discovery, *RREQ* message can be forwarded without waiting for verification to be processed (Zapata, 2006) as verification can be confirmed on the reply route. Such delayed confirmation of verification is not possible for the *RREP* message and certificates must be verified before the *RREP* message can be securely forwarded and trusted routes established. Therefore the problem is that the verification of the destination certificate $Cert_B$ may cause a delay in route establishment because $Cert_B$ is distributed with the *RREP* message.

A solution to this is the use of back tracked verification. If any intermediate node has $Cert_B$, it can distribute $Cert_B$ to the reverse route, during *RREQ* message propagation. When a *RREQ* message is forwarded a *flag* is appended identifying if the forwarder has the destination certificate $Cert_B$. Node *P* receives the *RREQ* message and updates the reverse route entry with $flag_{cert}$ indicating if the previous hop has $Cert_B$. Node *P* checks if it has $Cert_B$ in its certificate repository and assigns an appropriate value to *flag* before forwarding the *RREQ*. If node *P* has $Cert_B$ and the reverse route variable $flag_{cert}$ indicates that the previous hop does not have $Cert_B$ then *P* sends a unicast certificate message containing $Cert_B$. The $Cert_B$ is propagated along the reverse route by checking the routing table entry $flag_{cert}$ and responding in a similar fashion. This allows the destination certificate $Cert_B$ to be distributed during route discovery phase independent from route establishment. Therefore the neighbour and the originator certificate ($Cert_A$) are verified without causing any delay upon the route discovery. The destination certificate ($Cert_B$) is verified on the *RREP* message, and this delay is elevated by a prior distribution of $Cert_B$ on the reverse route where this is possible. Direct and indirect trust establishment is realised through the route establishment phase of the ad hoc routing scheme. During the initial stage of route establishment the network is flooded with routing requests and in turn certificate exchange messages. It can be expected that there will be a large packet overhead during the initial trust establishment stage.

3.0 DISCUSSION

3.1 Performance Evaluation

The performance of the proposed DITD protocol was analysed in a simulation study, to identify the effects of the certificate exchange and verification protocol upon the network

layer and network performance. Such analysis was taken under varied load, mobility, and number of nodes. The DITD was implemented in C++ as a network layer application for *ns-2* (release 2.31) (http://isi.edu), allowing key distribution and trust establishment in an on-demand manner.

3.2 Simulation Model

A wireless ad hoc network is simulated using the *ns-2* designed IEEE 802.11b physical layer and medium access control (MAC) protocols. The transmission range of each node was set to 250m. The network area is 1000m x 1000m. For the certificate exchange analysis the number of nodes was varied from 10 nodes to 60 nodes with 10 iterations and the number of connections was set to half the number of nodes in the network. The *ns-2* constant bit-rate (CBR) traffic generator was used to simulate the traffic load. The CBR packet size was set to 512 bytes and the traffic load was varied from 1 CBR pkt/s to 8 CBR pkts/s in increments of 1 CBR pkt/s. All the traffic sources are started before the initial 150 seconds, forcing maximum certificate distribution at the network establishment phase. The simulation time is set for a total of 1500 seconds.

Mobility models are a point of continued research however an optimum mobility model which simulates the realistic mobility of an ad hoc network is not readily available (Kim et al, 2009). The DITD protocol does not rely exclusively on mobility to achieve its goal. The DITD simulation uses the *setdest* model keeping with other existing literature using ns-2. Mobility was varied at three intervals 0.2m/sec, 5m/sec and 20m/sec simulating a network with low, moderate and rapid mobility. Pause time was set to 1 second.

For verification analysis the OpenSSL library is used for the security analysis of certificate verification. The size of the certificates was set to 450 bytes. ECC (Elliptic Curve Cryptography) is used. Choosing ECC is based on the expectation that it will be used in the future when more security is needed and accordingly longer keys must be used. As key size increases, the computational overhead for ECC increases at a much slower rate to other schemes like RSA Table 1 shows the times run on a Compaq iPAQ 3670 according to (Zapata, 2006).

Table 1: Times in milliseconds for Compaq iPAQ 3670

	RSA	DSA	ECC
Key length	1024	1368	160
Sign	210	90	42
Verify	6	110	160

The simulation's routing environment is a reactive, on-demand routing network and the Ad Hoc On-demand Distance Vector (AODV) routing protocol (Perkins *et al*, 2003) is used. The DITD model is implemented by modifying the AODV protocol in C++ providing the security model as a network layer application.

3.3 Simulation Results

The simulation results for the DITD protocol are presented and discussed aiming to assess the impact of the DITD protocol on the network performance. The certificate distribution mechanism and verification mechanism are analysed independently.

3.3.1 Certificate Distribution

Network performance can be analysed by the following metrics: packet delivery ratio (PDR) and the end-to-end packet delay. These metrics are set as functions of varied load, mobility and number of nodes. The network capacity and density are simulated by the varied number of nodes in the network. The analysis is isolated to the routing layer and routing control packets. The success rate of communication is represented by the PDR factor and the speed of the communication is represented by the end-to-end delay. We compare the proposed

DITD to a reference routing protocol AODV to investigate performance correlation between the two under differing load, node density, and mobility conditions.

In Figure 2(a) and Figure 2(b) it is observed that the DITD certificate distribution model has a strong correlation to the AODV reference simulation for PDR across varied load, mobility and number of nodes. Figure 3(a) and Figure 3(b) show that the DITD certificate distribution model adds no significant delay to the delivery for a varying load, mobility and number of nodes. The observation is made from these two graphs that the DITD model has negligible impact on the network performance. Figure 2(b) and Figure 3(b) show that as the number of nodes increases representing an increase in network capacity and node density, strong correlation with the reference AODV routing protocol is observed due the fully distributive nature of the network. Therefore the DITD model's performance is not affected by an increase in network capacity and node density. In Fig. 2(a) it is noted that as load increases the PDR decrease as expected, under more intense traffic conditions, but DITD maintains its correlation to the reference AODV simulation, under differing loads. Furthermore the DITD model is observed to have negligible impact on network performance for 0.2 m/sec, 5 m/sec, 10 m/sec, and 20 m/sec mobility. Mobility is exploited in DITD and it is observed in Figure 2(a) that as the mobility increases the correlation between the simulation models not only become stronger but DITD begins to outperform the AODV reference model, simulating a greater PDR at higher motilities such as 10m/sec and 20m/sec. Capkun (Capkun *et al*, 2006) proposes a security model which relies on mobility to distribute certificates in a more efficient manner. Capkun's proposal relies upon mobility for the delivery of certificates thus creating a security dependency upon mobility. In Figure 2(a) at 0.2 m/sec, representing a scenario with mobility that tends towards that of a stationary network, the reference AODV and proposed DITD simulations maintain strong correlation. This shows that DITD is not dependent on mobility, therefore breaking the dependency relationship between security and mobility but maintaining the benefits gained by increased mobility.

3.3.2 Trust Establishment Analysis

Investigations are made on trust establishment phase; this phase will concurrently follow the route establishment phase. The simulation forces all traffic to start in the first 150 seconds therefore most traffic will commence at the start of a network therefore routing and security set up overheads will be maximised at an initial setup phase. During the establishment phase the packets sent by DITD greatly exceed that of the reference AODV packets, with an average of 51% increase. This is due to the addition of certificate exchange packets, which increase proportionally as the number of nodes in a network increases. Figure 4 *s*hows for the full simulation time, an 11% average packet overhead for a varying number of nodes and varied load, which is lower than expected. For the trust establishment period a significantly higher average of 51% is observed.

The significant packet overheads for the initial phase are seen in Figure 4, creating an unavoidable increase in computation with an expected resultant computational delay. Despite this increase in computation the average end-to-end delay has negligible difference to the reference AODV routing protocol, as observed in Figure 3(a) and Figure 3(b). This is because of the fully distributive characteristics of DITD and its independency from the routing mechanism. In (Capkun *et al*, 2006) Capkun proposed a fully distributive certificate exchange scheme that also experiences excessive overheads at the initial security establishment stage. Capkun's approach relies on an initial setup stage to distribute all certificates amongst the network and upon completion of this stage; security would be available to the network. Tanabe's (Tanabe and Aida, 2007) security model proposed on the routing layer for an ad hoc on-demand scenario, expects similar delays for the delivery of trust. The flaw in this was a resultant initial window period of weakened security. The DITD model experiences similar packet overheads at the initial stage but allows for secure communication from the start of the network, without weakened security or a significant increase in delay during bootstrapping phase.

3.3.3 Postponed and Back Track Verification

To minimize the delay caused by verification, postponed and back tracked verification is performed in DITD. The simulations analyse the delays caused by ECC (Elliptic Curve Cryptography) verifications with a verification time of 160ms. Four methods are analysed: the general verification which verifies certificates immediately after they have been received; Zapata's method (Zapata, 2006) which uses delayed verification, verifying all the certificates after the route has been established; and lastly the DITD analysis, simulating DITD with and without back tracking.

Figure 5 shows the end-to-end delay of the four verification methods against the AODV protocol. The DITD model has the smallest delay of only 20ms excess, which is 80% smaller than Zapata's delayed verification method. This is achieved by minimising the verifications that result in a delay of route establishment. Verifications are rather performed independent to route discovery and confirmed at a later stage. Figure 6 shows that DITD has minimized the verifications which affect delay by almost a 1000% compared to the general verification method. To minimizing the delay, DITD relies upon verifications occurring behind the scenes, this approach is more computationally taxing than that of Zapata's. The total number of verifications preformed indicates the computational overheads of each method in Figure 6. This presents a trade off between delay and computational cost where Zapata's approach provides less computational cost to establish security while DITD provides faster secure route discovery. Therefore the application of DITD would be time-dependent networks.

4.0 CONCLUSION

In conclusion the Direct, Indirect Trust Distribution (DITD) model provides a key management model to realize certificate distribution and optimum certificate verification on the routing layer of a wireless ad hoc network. Routing packets are exploited by localized certificate exchanges providing direct trust and indirect trust by certificating chaining. Localising certificate exchange messages, removes trust dependency upon multi-hop routes which are vulnerable to collapse due to the dynamic nature of wireless ad hoc networks. In comparison, the packets sent by DITD show an 11% overhead on the overall simulation and a significant 51% overhead for the initial 150s trust establishment phase, compared to the reference routing model. Despite these overheads simulations show that the DITD model does not produce any significant delay or drop in packet delivery ratio for varied load, mobility and number of nodes (Figure 3). This is because of DITD's independence from the routing mechanism and fully distributive nature, allowing secure communication from the start of the network formation. DITD's postponed and back track verification mechanism helps minimize delays caused by computationally costly verification. The efficiency of DITD is tested by implementation and trace simulations in *ns-2*. It is concluded that the DITD uses local certificate exchanges and delayed certificate verification as an efficient way to provide a novel solution to trust establishment and the distribution of vital keying material in a wireless ad hoc network.

5.0 REFERENCES

Acs ,G., L. Buttyan, and I. Vajda. (2006), Provably Secure On-Demand Source Routing in Mobile Ad Hoc Networks, *IEEE Transactions on Mobile Computing,* vol. 5, pp. 1533-1546.

Awerbuch, B., R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens. (2008), ODSBR: An on-demand secure Byzantine resilient routing protocol for wireless ad hoc networks, *ACM Trans. Inf. Syst. Secur.,* vol. 10, pp. 1-35.

Capkun, S. Capkun, L. Butty, and J.-P. Hubaux. (2003), Self-Organized Public-Key Management for Mobile Ad Hoc Networks," *IEEE Transactions on Mobile Computing,* vol. 2, pp. 52-64.

Capkun, S., L. Buttyan, and J.-P. Hubaux. (2006), Mobility Helps Peer-to-Peer Security, *IEEE Transactions on Mobile Computing,* vol. 5, pp. 43-51.

Grandison, T. (2003), Trust Management for Internet Applications, Imperial College London.

Hu,Y.-C. Hu, D. B. Johnson, and A. Perrig. (2002), SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks, in *Proceedings of the Fourth IEEE Workshop on Mobile Computing Systems and Applications*: IEEE Computer Society.

Hu,Y.-C., A. Perrig, and D. B. Johnson. (2005), Ariadne: a secure on-demand routing protocol for ad hoc networks," *Wirel. Netw.,* vol. 11, pp. 21-38.

Kim, V. Sridhara, and S. Bohacek. (2009), Realistic mobility simulation of urban mesh networks," *Ad Hoc Netw.,* vol. 7, pp. 411-430.

Perkins, C. E., E. Belding-Royer, and S. R. Das. (2003), "Secure Ad Hoc On-demand Distance Vector (SAODV) Routing.

Perkins, C., E. Belding-Royer, and S. Das. (2003), *Ad hoc On-Demand Distance Vector (AODV) Routing*: RFC Editor.

Tanabe,M. and M. Aida. (2007), Secure communication method in mobile wireless networks," in *MOBILe Wireless MiddleWARE, Operating Systems, and Applications* Innsbruck, Austria: ICST.

Theodorakopoulos G., J. S. Baras. (2006), On Trust Models and Trust Evaluation Metrics for Ad-Hoc Networks, *IEEE Journal on Selected Areas in Communications,* vol. 24, pp. 318-328.

Zapata, M. G. (2006), Key management and delayed verification for ad hoc networks, *J. High Speed Netw.,* vol. 15, pp. 93-109.

Zhou, L., Z. J. Haas. (1999), Securing Ad Hoc Networks," *IEEE Network: special issue on network security,* vol. 13, pp. 24-30.

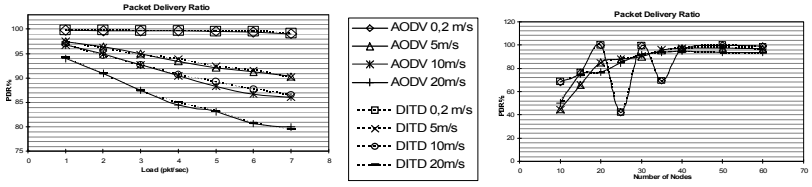

Fig. 2. DITD CBR packet delivery ratio (PDR) analysis for 0.2m/s, 5m/sec and 20m/sec mobility. *(a)* in varied load, 50 node environment, and *(b)* varied node density, single pkt/sec load environment.

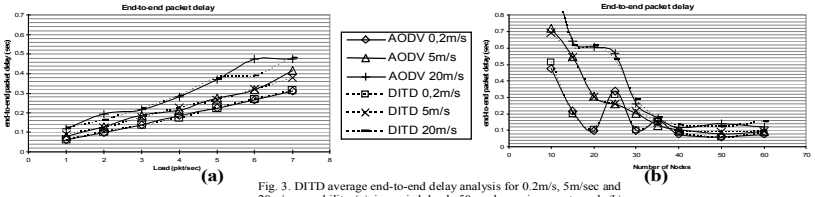

Fig. 3. DITD average end-to-end delay analysis for 0.2m/s, 5m/sec and 20m/sec mobility.*(a)* in varied load, 50 node environment, and *(b)* varied node density, single pkt/sec load environment.

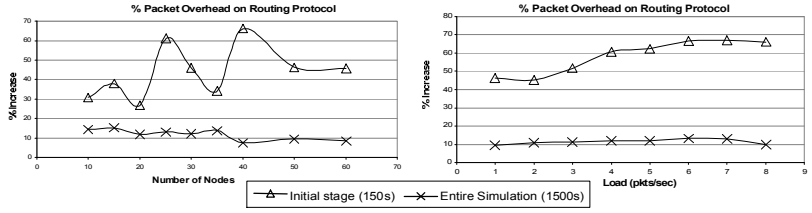

Fig. 4. DITD packet overheads for the initial stage and overall simulation time.

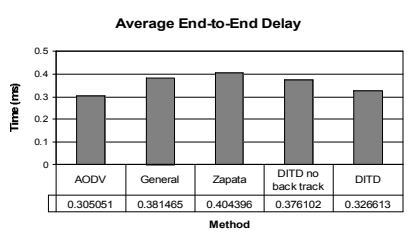

Fig. 5. Average end-to-end delay for 25 node network analysing verification delay

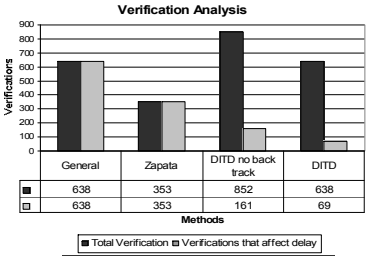

Fig. 6. Verification analysis for 25 node network