

# TECHNICAL RESEARCH REPORT

INORA- A Unified Signaling and Routing Protocol in  
Mobile AdHoc networks

*by D. Dharmaraju, A. Roy-Chowdhury, P. Hovareshti,  
J. S. Baras*

**CSHCN TR 2002-18  
(ISR TR 2002-33)**



*The Center for Satellite and Hybrid Communication Networks is a NASA-sponsored Commercial Space Center also supported by the Department of Defense (DOD), industry, the State of Maryland, the University of Maryland and the Institute for Systems Research. This document is a technical report in the CSHCN series originating at the University of Maryland.*

**Web site <http://www.isr.umd.edu/CSHCN/>**

# INORA - A Unified Signaling and Routing Mechanism for QoS Support in Mobile Ad hoc Networks

Dinesh Dharmaraju, Ayan Roy-Chowdhury, Pedram Hovareshti, J. S. Baras

{dineshd, ayan, hovaresp, baras}@isr.umd.edu

## Abstract

*Mobile Ad hoc NETWORKS (MANET) are characterized by bandwidth constrained wireless links, multiple hops and highly dynamic topologies. Providing QoS support in MANETs is hence a challenging task. This paper presents the design, implementation and evaluation of INORA, which is a network layer QoS support mechanism that makes use of the INSIGNIA in-band signaling mechanism and TORA routing protocol for MANETs. TORA provides multiple routes between a given source and destination. We present an effective coupling between TORA and INSIGNIA to get routes that are “best-able” to provide QoS requirements for a flow. INORA also combines congestion control with routing.*

**Keywords:** MANET, QoS, in-band signaling, resource reservation, soft-state, routing

## 1 Introduction

Mobile ad hoc networks (MANETs) are infrastructure-less networks of mobile nodes connected by wireless links. The topology of the network changes with time due to mobility of nodes. The channel conditions of the wireless medium are also time-varying. Hence providing quality of service (QoS) support for the delivery of real-time audio, video and data in MANETs presents a number of technical challenges.

The dynamic nature of MANETs calls for the soft-state reservation of resources across the network for providing QoS support, as against hard-state reservations in wired networks. The QoS support can be provided at the MAC layer (e.g. MACA/PR[1]) or at the network layer [2][3][4]. The network layer QoS support mechanisms can be broadly categorized into three classes, depending on the degree of coupling between the QoS resource management mechanisms and

the routing protocol.

**QoS routing** protocols search for routes with sufficient resources for the QoS requirements. These protocols work with the resource management mechanisms to establish paths through the network that meet end-to-end QoS requirements, such as delay or jitter bounds, bandwidth demand [5]. CEDAR[2] is an example of a routing protocol where the QoS provision mechanism is intrinsically tied to the routing protocol. However, QoS routing is difficult in MANETs. The overhead is too high for the bandwidth limited MANETs because there need to be mechanisms for a mobile node to store and update link information. Also due to the dynamic nature of MANETs, obtaining precise link information is very difficult. Moreover, the traditional meaning that the required QoS should be maintained once a feasible path is established is no longer true. The reserved resource may not be guaranteed because of the mobility-caused path breakage or power depletion of the mobile hosts.

**QoS signaling** is used to reserve and release resources, set up, tear down and renegotiate flows in the network. Such reservations can be both hard-state and soft-state. The latter is preferable in MANETs, because of the highly dynamic conditions in the network [6]. The signaling mechanism can be operated independent of the routing protocol. Here there is no interaction between QoS resource reservation mechanism and the routing protocol, the latter merely provides the route between the source and destination of a flow. The signaling protocol later establishes resources along the route chosen by the routing protocol. Here the routing protocol is completely decoupled from the signaling mechanism. INSIGNIA[3][6][7] is a good example for this.

An alternative to the above QoS signaling scheme would be to have a **coupling between the QoS mechanism and the routing protocol**. The coupling is looser than in QoS routing. The routing protocol provides a route between the source and destination of the flow.

The signaling mechanism provides feedback to the routing protocol regarding the route chosen and asks the routing protocol for alternate routes if the route provided doesn't satisfy the QoS requirements. The INORA (INSIGNIA+TORA) scheme that is presented in this paper belongs to this category. In INORA, INSIGNIA makes a callback to TORA[8] asking for alternate routes when the current route fails to meet the QoS requirements.

The remainder of the paper is organized as follows. In section 2 we give a brief overview of the INSIGNIA signaling system on which our scheme is based. The design and implementation of INORA is described in depth in section 3. Simulations of our scheme are presented in section 4, and we conclude with pointers to future work in section 5.

## 2 Overview of INSIGNIA signaling system

The INSIGNIA[6] in-band signaling system plays an important role in establishing, adapting, restoring and terminating end-to-end reservations for flows. INSIGNIA is designed to be light-weight in terms of the amount of bandwidth consumed for network control. It operates by setting up soft-state reservations for a flow across the path from the source of the flow to the destination in a mobile ad hoc network.

INSIGNIA uses the IP Options field in the IP header to convey the signaling information. The INSIGNIA fields are shown in figure 1. When a source node wants to establish a reserved QoS flow to a destination node, it sets the RES bit of the INSIGNIA IP option **service mode** of a data packet and sends the packet towards the destination. On reception of a RES packet, the intermediate nodes execute *admission control* to accept or deny the request. When a node accepts a reservation request, resources are committed and subsequent packets are scheduled accordingly. If the reservation is denied, packets are treated as *best effort mode* (BE) packets. The **payload type** field identifies whether the packet is of the type *base QoS* (BQ) or *enhanced QoS* (EQ). This is described in detail in [6]. The **bandwidth indicator** (BW\_IND) allows the QoS flow to specify its maximum (MAX) and minimum (MIN) bandwidth requirements for adaptive services. During request establishment, the bandwidth indicator reflects the resource availability at the intermediate nodes along the path between source-

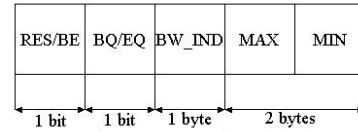


Figure 1: INSIGNIA IP options

destination pairs of different flows.

### 2.1 Admission Control

The source of a QoS flow sends out data packets with its service mode IP options field set to RES. All the intermediate nodes which receive packets with their *service mode* field set to RES perform admission control. At the first node where the admission control fails, the service mode is changed to BE (best effort).

Admission control failure can occur either when the node is unable to allocate at least the minimum required bandwidth ( $BW_{min}$ ) for the flow, or there is congestion at the node, i.e. the queue-size at the node has exceeded a threshold ( $Q > Q_{th}$ ).

### 2.2 QoS Reporting

QoS reporting is used to inform source nodes about the current status of the flows. Destination nodes actively monitor the current flows, inspecting the *status information* (e.g. bandwidth indicator) and *measured delivered QoS* (e.g. packet loss, throughput etc.). Although the QoS reports are normally generated periodically according to the sensitivity of application, QoS reports are sent immediately when required. The source, on reception of a QoS report indicating a flow degrade from reserved to best effort, may downgrade the flow. Here the feedback is end-to-end from the destination to the source.

## 3 INORA

INSIGNIA does not take any help from the network with regard to redirecting the flow along routes which are able to provide the required QoS guarantees. In INORA we make use of feedback on a per-hop basis to direct the flow along route that is able to provide the QoS requirements of the flow. We make use of the INSIGNIA in-band signaling system and TORA routing

protocol in the INORA scheme.

TORA operates by creating a *Directed Acyclic Graph* (DAG) rooted at the destination. The DAG is extremely useful in our scheme since it provides multiple routes from the source to the destination. We use this routing structure to direct the flow through routes that are able to provide the resources for the flow according to the QoS requirements of the flow. INORA can be classified into two schemes; we call one the coarse feedback scheme, and the other the fine feedback scheme. The two are described in the following subsections.

### 3.1 Coarse Feedback Scheme

The operations of the coarse-feedback scheme of INORA are illustrated through the following example.

Consider a QoS flow being initiated with node 1 as the source and node 5 as the destination.

1. Let the DAG created by TORA be as illustrated in figure 2. Let  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$  be the path chosen by the TORA routing protocol.
2. INSIGNIA tries to establish soft-state reservations for the QoS flow along the path. Node 4 is the first node at which admission control for the flow fails (because of either of the conditions mentioned in section 2.1). Node 4 sends an out-of-band *Admission Control Failure* (ACF) message to its previous hop (node 3) (figure 3).
3. Node 3 realizes that the next hop node 4 is not good for the current flow and re-routes the flow through another downstream neighbor (node 6) provided by TORA (figure 4).
4. If node 6 is able to admit the flow, the flow gets the required reservations all along the path. The new path would be  $1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 5$  (figure 4).
5. If node 6 is unable to admit the flow, it sends an ACF message to node 3 (its previous hop) (figure 5).
6. Node 3 realizes that it has exhausted all the downstream neighbors that it was provided by TORA. So, it sends a *Admission Control Failure* message to its previous hop (node 2), indicating that none of its downstream neighbors can accommodate the flow (figure 6).

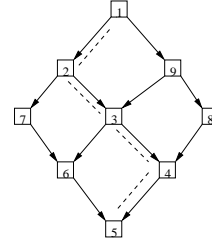


Figure 2: INORA Coarse Feedback: node 4 is a bottleneck; admission control fails at node 4

7. Node 2 now tries with its other downstream neighbors for the possibility of a path that can give the required reservations to the flow.

Note that as a result of this scheme, it is possible that different flows between the same source and destination pair can take different routes - as can be seen from figure 7, that to go from node 1 to node 5, flow 1 takes the path  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$  and flow 2 takes the path  $1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 5$ .

While INORA is trying to find a good route for the flow following admission control failure at an intermediate node, the packets are transmitted as *best effort* (BE) packets from the source to the destination. It should also be noted that there is no interruption in the transmission of a flow that has not been able to find a route in which resources have been reserved all the way from the source to the destination. Because of the nature of the *Directed Acyclic Graph* (DAG), INORA tries to get a route which satisfies QoS requirements locally. When this fails, the search for a route which satisfies the QoS requirement becomes more global. In the worst case, we would have searched the entire DAG for a QoS route. The state introduced in the nodes due to this search is soft. So, there is no overhead in maintaining the state. This search, which goes on in the background does not affect the delivery of the packets of the flow. Also, the scope of search for the routes is the DAG. INORA only chooses an appropriate route from the set of routes given by TORA. It does not trigger any route-querying mechanism to find routes which will satisfy the QoS requirements. The philosophy of INORA is that it tries to find better routes for a flow while the transmission of the flow continues without interruption.

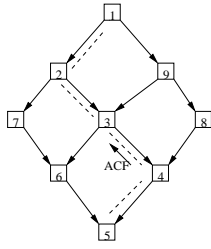


Figure 3: INORA Coarse Feedback: node 4 sends out-of-band ACF to previous hop (node 3)

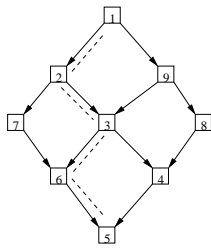


Figure 4: INORA Coarse Feedback: node 3 redirects the flow to node 6

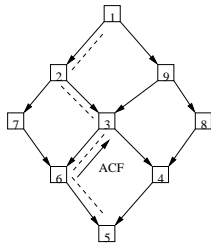


Figure 5: INORA Coarse Feedback: if node 6 fails to admit the flow, it sends ACF message to node 3

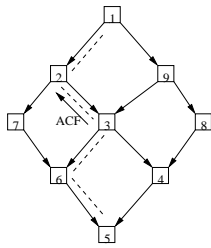


Figure 6: INORA Coarse Feedback: node 3, having exhausted all its next-hops, sends a ACF to its previous hop node 2

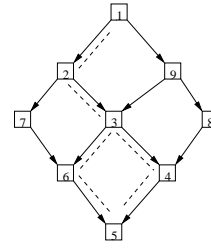


Figure 7: INORA Coarse Feedback: different flows between same source-destination pair can take different routes

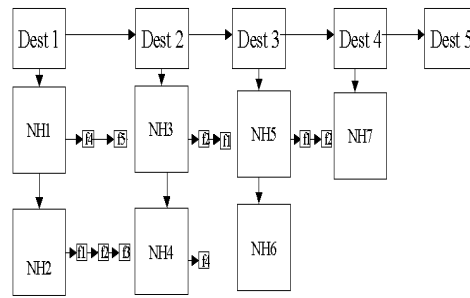


Figure 8: TORA Routing Table in INORA

### 3.1.1 Implementation Details

When a node X receives an Admission Control Failure (ACF) message from its downstream neighbor Y, it blacklists the downstream neighbor Y. Associated with the blacklist entry is a timer, which makes sure that the downstream neighbor Y is blacklisted long enough. The node Y must be blacklisted for the expected period of time required by INORA to search for a QoS route. This time is chosen according to the size of the network.

The TORA routing table is restructured in INORA as shown in figure 8. Associated with every destination, there is a list of next hops which is created by TORA. With the feedback that TORA receives from INSIGNIA in INORA, TORA associates the *next-hops* with the flows they are suitable for. So a routing lookup in INORA is based on the ordered pair (*destination, flow*). If TORA does not have the information about the best route for the given flow, the routing lookup is just based on the destination. In that case, TORA gives the downstream neighbor with the least *height* metric[8]. If any of the nodes is not INORA-aware, normal operations of INSIGNIA and TORA continue.

### 3.2 Class-based Fine Feedback Scheme

In this scheme, we divide the  $(BW_{min}, BW_{max})$  interval into  $N$  classes, where  $BW_{min}$  is the minimum bandwidth required by a QoS flow and  $BW_{max}$  is the maximum bandwidth required by the QoS flow. The IP options field in the IP header which carries the IN-SIGNIA information, now carries an additional *class* field. This field signifies the amount of bandwidth that has been allocated for the flow along the path. The operation of the protocol is illustrated by the following example.

Consider a QoS flow being initiated with node 1 as the source and node 5 as the destination, with minimum bandwidth requirement  $BW_{min}$  and maximum bandwidth requirement  $BW_{max}$ . Let the flow be admitted with class  $m$  ( $m < N$ ) at node 1.

1. Let the DAG created by TORA be as shown in figure 9. Let  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$  be the path chosen by the routing protocol.
2. INSIGNIA tries to establish soft-state reservations for the QoS flow along the path.
3. Node 2 is able to admit the flow with class  $m$  as was requested by its previous upstream hop node 1.
4. Suppose that node 3 has admitted the flow with class  $l$ , but has not been able to allocate the bandwidth of class  $m$  ( $l < m$ ), as requested by its previous hop 2 (figure 10).
5. Node 3 now sends an *Admission Report* message ( $AR(l)$ ) to the upstream previous hop (node 2), indicating its ability to give class  $l$  bandwidth to the flow (figure 10).
6. Node 2 splits the flow in the ratio of  $l$  to  $m - l$  and forwards the flow to node 3 and node 7 respectively, in that ratio. This means that the flow of class  $m$  has been split into two flows of class  $l$  and  $m - l$  and is forwarded to nodes 3 and 7 respectively (figure 11).
7. Suppose that node 7 is unable to give class  $(m - l)$  as requested by the upstream previous hop 2, but is only able to give class  $n$  ( $n < m - l$ ). 7 sends an *Admission Report* message ( $AR(n)$ ) to the upstream previous hop, node 2 (figure 12).

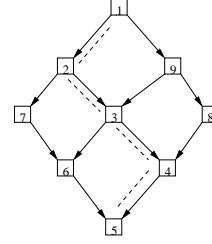


Figure 9: INORA Fine-Feedback: node 3 has admitted the flow with class  $l$ , but is not able to give the bandwidth-class that node 2 (previous hop) is able to give, say  $m$ ,  $m > l$ .

8. Node 2, realizing that its downstream neighbors have been unable to give the class  $m$ , which it had requested, informs its ability to give a class  $l + n$  ( $l + n < m$ ) by sending a *Admission Report*  $AR(l + n)$  to its previous hop node 1 (figure 13).
9. Node 1 now tries to find another downstream neighbor, which might be able to accommodate the flow with class  $(m - (l + n))$ .

Note that when a node is unable to admit a flow, either because it cannot allocate the minimum bandwidth  $BW_{min}$  required by the flow, or due to congestion at a node, it sends *Admission Control Failure* messages as in the coarse-feedback scheme described in section 3.1. So, the fine-feedback scheme includes the features of the coarse-feedback scheme. The fine-feedback scheme, like the coarse-feedback scheme first tries to search for a QoS route, which can give the requested bandwidth class locally. The search becomes more global if it is not able to find locally the QoS route which gives the required cumulative class. Also, it is worth noting that a single flow can get split, and the packets can take different routes from the source to the destination (figure 14). This can result in packets being received out of order at the destination. The real-time applications with QoS requirements typically use RTP as the transport protocol. RTP does re-ordering of the packets. If TCP is used as the transport protocol, packets arriving out of sequence can trigger TCP's congestion avoidance mechanisms. The effect of out-of-order delivery on TCP has to be further investigated.

#### 3.2.1 Implementation Details

Consider the example mentioned in 3.2. When node 2 receives an  $AR(l)$  from node 3 and  $AR(n)$  from

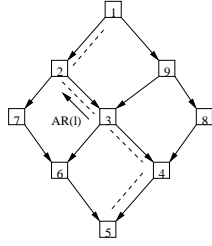


Figure 10: INORA Fine-Feedback: node 3 sends Admission Report  $AR(l)$  to node 2.

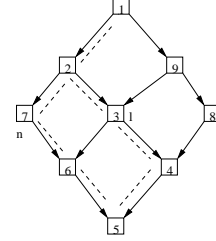


Figure 14: INORA Fine-Feedback: a single flow gets split and takes different paths to the destination.

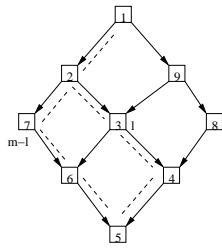


Figure 11: INORA Fine-Feedback: node 2 splits the flow between the next hops, 7 and 3 in the ratio  $m - l$  to  $l$  respectively.

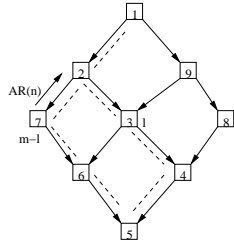


Figure 12: INORA Fine-Feedback: node 7 is unable to give  $m - l$ , but only  $n < m - l$ . It sends  $AR(n)$  upstream.

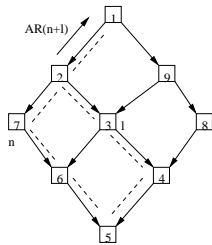


Figure 13: INORA Fine-Feedback: node 2 sends  $AR(n + l)$  indicating the bandwidth that it can support.

Table 1: Average delay of QoS packets

QoS Scheme	Avg. end-to-end delay (sec)
No feedback	0.049
Coarse feedback	0.043
Fine feedback	0.034

Table 2: Average delay of all packets(QoS/non-QoS packets)

QoS Scheme	Avg. end-to-end delay (sec)
No feedback	0.108
Coarse feedback	0.021
Fine feedback	0.062

Table 3: Overhead in INORA schemes

QoS Scheme	No. of INORA pkts/data pkt
Coarse feedback	0.0174
Fine feedback	0.1833

node 7, indicating the ability of the downstream neighbors to give classes  $l$  and  $n$  to the flow respectively, as against the requested class  $m$  ( $l + n < m$ ), node 2 makes a note of the class that each downstream neighbor has been able to allocate in the *Class Allocation List* and associates timers with those entries. The TORA routing tables here are similar to the *coarse-feedback scheme* as illustrated in figure 8. There is an additional *class* field in the *flow* entries of the routing table. The routing table look-ups are made on the basis of the ordered 3-tuple (*destination, flow, class<sub>req</sub>*) where *destination* stands for the destination for which we are looking up routes, *flow* stands for the flow for which we are looking up routes and *class<sub>req</sub>* stands for the bandwidth class requested by the flow.

## 4 Simulations

We performed *ns-2* simulations to evaluate the INORA framework. The INSIGNIA code was obtained from the COMET group at Columbia University [9]. The TORA *ns-2* code was taken from the Center for Satellite and Hybrid Communication Networks, University of Maryland[10]. We made modifications to the INSIGNIA and TORA source codes to incorporate the INORA scheme. For the wireless environment, CMU Monarch wireless extensions [11] for *ns-2* were used. We ran experiments with the INORA schemes (*coarse-feedback* and *fine-feedback*), and also the original INSIGNIA and TORA, running independent of each other without feedback. In the INORA *fine-feedback scheme*, we choose the number of classes to be 5 ( $N = 5$ ).

Our simulation scenario is a  $1500m \times 300m$  rectangular grid with 50 mobile nodes. The transmission range of a node is  $250m$ . The node mobility follows *Random Way-point Model*. The nodes move with speeds uniformly distributed between 0-20m/s. We have 10 flows, 3 of which have QoS requirements and the remaining 7 flows don't have QoS requirements. The packets (belonging to both QoS and non-QoS flows) are of length 512 bytes. The sources generate CBR traffic. The non-QoS flows have an inter-packet interval of 0.1sec. (corresponding to a bandwidth  $BW = 40.96 kb/s$ ). The QoS packets have an inter-packet rate of 0.05sec (corresponding to a bandwidth of  $81.92 kb/s$ ). The QoS flows request for a reservation of  $BW_{min} = BW = 81.92 kb/s$ , and  $BW_{max} = 2BW = 163.84 kb/s$ .

## 4.1 Results

We evaluate the performance of the INORA schemes by observing the end-to-end delay of the packets and the control message overhead.

The average end-to-end packet delay for the QoS flows in different schemes is as shown in Table 1. We see that the INORA *coarse-feedback* has lesser average delay than INSIGNIA and TORA operating without feedback. The INORA *fine-feedback* scheme performs better than the INORA *coarse-feedback* scheme. This is because the INORA feedback schemes try to find paths which can allocate the requested bandwidth reservations to the QoS flows. The *fine-feedback* scheme does this in a much fine-grained manner when compared to the *coarse-feedback* scheme. So, we have the *fine-feedback* scheme performing better than the *coarse-feedback* scheme.

Table 2 shows the average end-to-end delay experienced by all the packets (from both QoS and non-QoS flows). We see that the INORA feedback schemes perform better than INSIGNIA and TORA operating without feedback. It can be seen that the average delay is reduced by 80% from the no feedback case in INORA *coarse-feedback* scheme. By trying to find the paths which can allocate the required bandwidth to the flows and by performing load balancing in the network, the INORA schemes ensure that the overall congestion in the network is reduced. So, we have lesser end-to-end delay for packets. We find that INORA *fine-feedback* scheme has higher average end-to-end delay (for QoS and non-QoS packets) when compared to INORA *coarse-feedback* scheme. This is because the INORA *fine-feedback* scheme does fine-grained feedback (by splitting the QoS flows along different paths), which benefits the QoS flows more at the cost of the non-QoS flows.

Table 3 shows the overhead in the INORA schemes. We find that the number of INORA control messages transmitted per every QoS data packet delivered is more for the *fine-feedback* scheme as compared to the *coarse-feedback scheme*. This is as expected because of the additional *Admission Report* messages for fine-grained control in the *fine-feedback* scheme.

## 5 Conclusions and Future Work

In this paper INORA, a QoS support mechanism using INSIGNIA in-band QoS signaling system and TORA



routing protocol for ad hoc networks, has been proposed. The implementation and an evaluation of INORA has also been presented. We have shown by simulations that INORA schemes do well by trying to find paths in the network that can allocate the requested bandwidth requirements. By performing load-balancing in the network, they also aid the delivery of non-QoS flows.

We plan to do further work with INORA in congestion control. In wireless networks, congestion at a wireless node is related to congestion in its one-hop neighborhood. We intend to incorporate a suitable mechanism in INORA to reflect this fact, so that congested neighborhoods can be avoided by QoS flows. The effect of out-of-sequence delivery on TCP in the INORA coarse-feedback scheme should also be investigated.

## 6 Acknowledgements

The authors would like to thank Dr. Scott Corson, Dr. John Baras, Dr. Richard La, Matthew Impett, Ulas Kozat for their support in making this work possible.

## References

- [1] C.R. Lin and M. Gerla, "MACA/PR: An Asynchronous Multimedia Multihop Wireless Network," *Proceedings of IEEE INFOCOM '97*, 1997.
- [2] P. Sinha, R. Sivakumar, V. Bhargavan, "CEDAR: a Core-Extraction Distributed Ad hoc Routing Algorithm," *IEEE INFOCOM '99*, New York, NY.
- [3] S-B. Lee, G-S. Ahn, X. Zhang, A. T. Campbell, "INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks," *Journal of Parallel and Distributed Computing*, Vol 60 No 4, April 2000
- [4] S. Chen, and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad Hoc Networks," *IEEE Journal on Special Areas in Communications*, Vol. 17, No. 8, August 1999.
- [5] K. Wu, J. Harms, "QoS Support in Mobile Ad Hoc Networks," *Crossing Boundaries- the GSA Journal of University of Alberta*, Vol. 1, No, 1, Nov. 2001, pp.92-106
- [6] S-B. Lee, A.T. Campbell, "INSIGNIA: In-band signaling support for QoS in mobile ad hoc networks", *Proc of 5th International Workshop on Mobile Multimedia Communications (MoMuC, 98)*, Berlin, Oct. 1998
- [7] G-S. Ahn, A.T. Campbell, S-B. Lee, X.Zhang, "INSIGNIA," *Internet Draft*, draft-ietf-manet-insignia-01.txt, Oct 1999
- [8] V. Park, S. Corson, "Temporally Ordered Routing Algorithm(TORA) version 1 functional specification, draft-ietf-manet-tora-spec-04.txt," July 2001
- [9] INSIGNIA *ns-2* source code from Comet Group, Columbia University URL:[http://comet.ctr.columbia.edu/ns\\_source\\_code.html](http://comet.ctr.columbia.edu/ns_source_code.html)
- [10] TORA *ns-2* source code from CSHCN, University of Maryland, College Park. URL:<http://www.cshcn.umd.edu/tora.shtml>
- [11] Monarch *ns-2* wireless extensions from Monarch group, Carnegie Mellon University. URL:<http://www.monarch.cs.cmu.edu/cmuns.html>