A QoS Solution to Hard Real Time Delay Satisfaction in Ad Hoc Networks

Alok Kumar Jagadev, Binod Kumar Pattanayak*, Manoj Kumar Mishra, Manoj Ranjan Nayak

Siksha 'O' Anusandhan University, Jagamohan Nagar, Khandagiri, Bhubaneswar, Orissa, India, Postal Zip : 751030
*Corresponding Author

Summary
Due to mobility features of Ad Hoc networks, it becomes too difficult to guarantee delay requirements for real time applications across such networks. However, such real time guarantees can be optimally achieved with a reasonable presumption on node mobility, network traffic and reliability with predictable features. In this paper, we present a routing algorithm to achieve the guarantees for Hard real time network traffic, based on Dynamic Source Routing (DSR). It implements a node resource reservation policy along each traffic path. Admission control is performed with preservation of reserved node resources and is adapted to node mobility ultimately preventing path breaking. Our routing protocol is implemented under network simulator ns2 and simulation results are demonstrated with performance evaluation.
Keywords : Expiration delay, real time, QoS, mobility

1. Introduction

Wireless communication in ad hoc networks, its mobility feature and ease in deployment have found wide implementation in strategic applications like military service and industrial applications too [27]. Such infrastructure-less networks include devices (nodes) in a wireless communication environment, which can, at the same instance, work as a sender, receiver or even as a router. These devices can move in the network with frequently changing topologies, consequently causing no impairments in communication. However, appropriate routing protocols should be implemented in ad hoc networks to make the communications consistent between the sender and the receiver.
Mobility nature of ad hoc network makes it difficult to chose a proper and efficient routing protocol. The routing protocols used in wired networks can never be suitable for mobile ad hoc networks.(MANETs). A wide range of routing protocols have already been proposed for MANETs by different authors, which are globally classified into two categories : proactive and reactive.
OSLR (Optimized State Routing Protocol), a proactive protocol [12], is capable of maintaining a consistent route of transmission across a MANET. The same is achieved with periodical updations of the routing table to all possible destinations. On the other hand, reactive protocols, implement on- demand route discovery by broadcasting a route request message through the network, where a discovered path is maintained as long as it is desired. In reactive protocol DSR (Dynamic Source Routing) [24], the source transmits a route request message to destination and waits for a route reply message, on receiving which a route is selected. The source transmits the required traffic along the selected path.
The above mentioned protocols successfully determine the paths between sources and destinations. However, these protocols do not guarantee the QoS requirements to bandwidth management and delay management in ad hoc networks. QoS support to bandwidth and delay requirements becomes essential for real time applications. Authors in [7], [9] have proposed methods of routing with QoS in ad hoc networks.
The architectures Intserv (Integrated Services) and Diff Serv (Differentiated Services) serve to the base of QoS routing protocols. The IntServ approach adopted in BRuIT (Band width Reservation Interferences influence) [8] protocol implements a node reservation principle along a selected path in an ad hoc network, whereas, the DiffServ approach relies on classification of traffic in ad hoc networks. However, real time traffic should be allowed on a higher priority basis so as to reach the specified destination in shortest possible time, as implemented in MQDR (Multipath QoS routing Protocol Supporting Diffserv) Protocol.
However, such protocols are not applicable to hard real time traffic. In fact, support to QoS is not sufficient, rather it must be guaranteed. In case the real time constraints, like a specified deadline to communication, are not maintained, it may result in unpredictable consequence and in some cases may lead to system crash. Hence, hard real time traffic need the deadline QoS constraints be met.
In this paper, we address real time solutions to allow real time packets be routed with respect to their deadline constraints. Due to lack of centralized coordination and mobility features of ad hoc networks, guarantee to deadline constraints becomes more challenging. As there is no centralized coordination, it becomes difficult to provide guarantee for real time packets to reach the destination before the deadline.
In addition, new real time traffic may interfere with previous real time traffic and prevent the packets from reaching their destinations before deadline. We address this problem with a proposed admission control scheme to

Manuscript received December 5, 2009
Manuscript revised December 20, 2009

maintain the guarantee of the accepted real time traffic. Thus, a new traffic can be accepted only in case it can be transmitted with respect to its specified deadline and thereby not disrupting the guarantees of previous traffic. In this scheme, admission control is performed in collaboration with all nodes of a selected traffic path rather than an independent admission control.

Mobility becomes another challenging issue, where the path to a destination should be consistently maintained even if some links break down. For this reason, we must be able to predict such link failures in advance and establish another path for the traffic with respect to the real time constraints.

In view of the specified admission control and mobility prediction, we suggest a new protocol RTD-DSR (Real Time DSR Protocol with Delay constraints), incorporated with reactive protocol DSR, which helps to determine paths with real time requirements. This protocol is implemented with network simulator ns2 to verify its efficiency for real time traffic.

The following part of the paper is organized as follows. In section 2, we illustrate a brief overview of different solutions to QoS in ad hoc networks. Section 3 covers the different aspects of routing protocol DSR. Our proposal based on expiration delay is discussed in section 4. Section 5 includes a discussion on mobility features. Our proposal is evaluated through simulations in section 6. In section 7, we conclude the paper with positive enhancements to our protocol, and in section 8 we suggest the future work.

2. QoS in Ad Hoc Networks

Some real time applications strictly rely on the quality of service (QoS) requirements like end-to-end delay, deviation of latency (jitter), bandwidth consumption and reliability in communication.

Two widely implemented architectures, namely, IntServ and DiffServ, are designed to provide QoS requirements to flows across a network. In a network supporting IntServ (Integrated Services) architecture, all the routers need to implement IntServ and every application in the network that requires some guarantees, has to make an individual reservation. RSVP (Resource Reservation Protocol), incorporated in IntServ, enables the router to make decisions on support to requested reservation for a flow. DiffServ (Differentiated Services) [13] principally operates on the principle of traffic classification, where each data packet may belong to one or more classes, and can be treated differently by routers depending on the priority of its class.

Several approaches have already been proposed with IntServ and DiffServ for providing QoS in the literature. Some of the routing protocols for Ad Hoc networks have been extended to support QoS requirements, whereas some other protocols are independently designed for QoS alone. Of the above, some QoS Ad hoc routing protocols are specifically designed for real time application.

2.1. Routing protocols with extensions for support to QoS

In view of providing QoS solutions in Ad hoc networks, researchers [8] have incorporated enhancements in existing Ad hoc routing protocols to support QoS requirements. OLSR, a proactive routing protocol has been enhanced to QOLSR(QoS in OLSR) by the researchers [9]. In QOLSR, each node in the ad hoc network needs to declare its multipoint relay sectors (MPR) that represent a subset of links with its neighbors. These MPRs can retransmit the packets during flooding or broadcasting procedures, which consequently reduces retransmissions. The signaling messages of OLSR are modified in QOLSR with the addition of additional fields which hold QoS conditions, and thus, QOLSR supports multiple-metric routing criteria.

Reactive ad hoc routing protocols like AODV and DSR, which implement on-demand route selection policy, have already been enhanced to support QoS in ad hoc networks. Enhanced version of AODV, i.e. AODV with QoS [7], incorporates a QoS object extension in the signaling messages, which specifies bandwidth and delay parameters, where a node can react to a signaling packet, only if it can satisfy the specified requirements.

DSR, a reactive routing protocol, is enhanced to MP-DSR[11], which provides multiple paths after the Route Discovery phase, which consequently guarantees end-to-end reliability. Thus, the data packets can be transmitted over discovered paths which satisfy the reliability requirements, as specified by the sender.

2.2. Routing Protocols with QoS

Routing Protocols with QoS can be classified into three categories:

i) routing protocols based on IntServ;
ii) routing protocols based on DiffServ;
iii) routing protocols combining both IntServ and DiffServ.

Protocols based on IntServ architecture include protocols like CEDAR (Core Extraction Distributed Ad Hoc Routing) [8], in which a set of nodes in the ad hoc network is elected to form a core group, which acts as a backbone for communication. Information about available bandwidth along stable links in the network is propagated to all core nodes, which consequently compute the paths for communication.

CBMP (Centralized Bandwidth Management Protocol) [1], proposed by us and based on IntServ architecture, principally relies on a centralized Bandwidth Manager (BM), for bandwidth allocation as per the QoS requirements of a flow. It implements on-demand bandwidth allocation policy. Architecture of CBMP includes a per flow Rate Adaptor (RA), which requests BM for bandwidth allocation for the flow. However, CBMP can be effectively used for single-hop ad hoc networks, and needs enhancement for implementation in multi-hop ad hoc networks.

IntServ architecture based protocol AQDR (Adhoc QoS On-demand Qouting) [8] also includes some of the mechanisms to allow QoS-routing. In this protocol, admission control is performed for new flows depending on available resources and provides fast recovery on QoS violations, where in case of QoS violation, the destination broadcasts a "route update message" back to the source, if it detects QoS violation along the existing active route. In response to the "route update message", the source re-directs the traffic along the path of the first update message.

Another IntServ architecture based protocol, BRuIT (Bandwidth Reservation under Interferences influence) [9] adopts the principle of bandwidth reservation. As per this approach, a node requires to broadcast a route request message for reservation of bandwidth for a flow, with the address of the receiver and the amount of required bandwidth to be consumed. On receiving this message, each node verifies if it has enough bandwidth to handle the reservation. If the available unused bandwidth appears to be enough for the reservation at a node, then it forwards the route request message to its neighbors, discards the request otherwise. If the route request message reaches the destination successfully, the destination sends a route reply message back to the source. Each node on receiving the route reply message reserves the required amount of bandwidth for the flow and consequently decreases its free bandwidth counter by the required amount [17].

DiffServ architecture based routing protocol, courtesy piggybacking [8], implements piggybacking of low priority traffic into the high priority traffic, if there is enough space for it. Another DiffServ based routing protocol MQRD (Multipath QoS Routing Protocol of supporting DiffServ) [7] is also designed for supporting QoS, which adopts multipath routing policy with avoidance of traffic congestion and link failure. With the help of DiffServ, MQRD divides traffics into different priority levels, which helps in congestion avoidance. Different mechanisms of scheduling and queuing in these protocols helps to support routing. FQMM (Flexible QoS Model for MANETs) belongs to third type of protocols, which combines both IntServ and DiffServ principles. It provides QoS requirements to different traffics depending on the priority levels, where high-priority traffics are handled by IntServ and other traffics by DiffServ. Thus, FQMM implements a service differentiation policy and provides QoS to classes of traffic with specific QoS constraints, and explores the possibility of reservation of sufficient resources for a flow [22].

2.3.QoS Routing protocols for Real-time applications :

The challenges and constraints in real time data transmission across ad hoc networks need Qos routing solutions. Researchers, in providing such solutions, have mostly adopted DiffServ principle oriented around classification of flows [11]. SWAN (Service Differentiation in Stateless Wireless Ad hoc Networks) [14], which implements a stateless network model based on feedback information from the network, presents such a routing solution. With this feedback information, the distributed control algorithms, incorporated in SWAN, provide a service differentiation policy based on the class of the traffic. The principle of the service differentiation in SWAN is based on delaying the best effort traffic as per the requirements of real time traffic in order to support network conditions.

Another QoS routing protocol, QPART (QoS Protocol for Ad hoc Real-time Traffic) [11], provides probabilistic QoS guarantees to real-time applications in mobile ad hoc networks. It relies on scheduling the packet of a flow as per its QoS requirements. Such QoS requirements may include end-to-end delay for delay-constrained flows, throughput for bandwidth-constrained flows, and reliability for best effort flows. In QPART, a real-time flow possesses its own packet queue, contention window and has independent access to the communication channel like an independent node. The size of contention windows are specified at MAC layer. In this approach, traffic of higher priority is assigned with a smaller size of contention window so as to increase the probability of access to the channel before other traffics. Sizes of contention windows can be regulated until the network can no longer support real-time traffic requirements. Thus, traffic with a lower priority is rejected with a higher probability.

The QoS routing protocols discussed above, do not provide solutions for hard real time applications. In fact, these protocols address QoS constraints and do not provide any guarantee to such constraints. Some of the protocols focus on improving the quality of transmission by adding QoS conditions such as QOLSR and AODV with QoS, or by providing better availability like MP-DSR, whereas some other protocols are useful for reservation of resources for flows like BRuIT.

Protocols like SWAN and QPART are not suitable for hard real time applications, as they adopt DiffServ architecture : a) DiffServ [4] doest not provide End-to-End QoS, as it does not maintain a per flow state information; b) as DiffServ does not adopt any reservation of resources for flows, probability of congestions increases. Hence, in our implementation, we use the advantage of IntServ architecture like quantitative specification of the resource requirements of a flow.

Thus, in our current research work, we propose a new hard Real time routing protocol, which can successfully guarantee the real time constraints. In view of this, we extend an existing ad hoc routing protocol to incorporate real time constraints adopting the principles of IntServ architecture. For the purpose, we use DSR routing protocol, as it provides excellent performance for routing in multi-hop mobile ad hoc networks, as suggested and proved by the authors in [18].

3. Dynamic Source Routing (DSR) protocol

In reactive ad hoc routing protocol DSR [13,3,24), two major phases are incorporated for successful multi-hop routing : Route Discovery and Route Maintenance. Further, the phase of Route Discovery is split into two sub phases : route request & route reply.

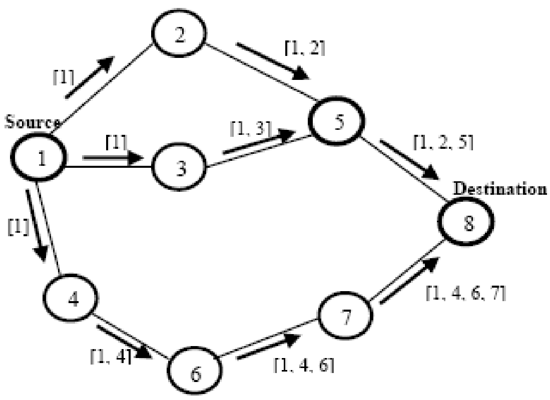


Fig. 1 : Broadcasting route request packet

As depicted in Fig.1, node 1 requires to send data to node 8, from which it broadcasts a route request packet to its neighboring nodes. Consequently each neighboring node (nodes 2,3,4), on receiving the route request packet, adds its own address to the route record of the packet and forwards to its neighbors. This process is repeated at each of the intermediate nodes until the route request packet reaches the destination or one of the intermediate nodes finds a valid path to the specified destination in its route cache. The destination node (node 8), on receiving the first route request packet, reverts the route and sends a route reply packet. For example, in Fig. 1, node 8 receives the first route request packet from node 5 with recorded route (1→2→5), following which it sends the route reply packet back to source node (node 1) along the route (5→2→1).

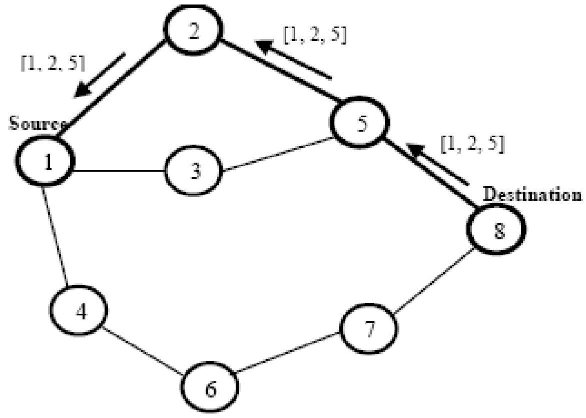


Fig. 2 : Sending a Route Reply back to the source

During the transmission of the traffic, if one of the links along the route breaks, a route error packet is generated by the corresponding intermediate node and sent back to the sender. This process of notification is handled by the route maintenance phase.

Real time transmission in DSR is not applicable for the following two reasons : (1) since DSR does not provide the guarantees for delay constraints, packets cannot be ensured to reach the specified destination before the required deadline; (ii) In DSR, the mobility issues are not well maintained, e.g. in case of a link failure, an intermediate node, on detecting the failure, sends an error packet back to the source, which is too time-consuming, and real time transfers do not accept such a delay.

However, DSR provides a set of advantages, which made us to chose it as our base protocol for our current work. These advantages are (i) DSR, which is a reactive protocol, is preferable for real time transfer, as at any point of time it keeps the constructed route up-to-date; (ii) simplicity and flexibility features of DSR facilitates its implementation of our real time extension; and iii) a route request packet broadcasted during route request phase can incorporate the real time constraints. Hence, it is our endeavor to extend IntServ based DSR to incorporate real time applications, which would support specifically hard real time transfers.

4. Our proposal based on expiration delay

The principal goal behind our proposal is to provide a solution for hard real time flows to be transmitted through ad hoc networks correctly and in time. Our proposal is oriented around a solution based on expiration delay to deadline (real-time constraint) in order to ensure the delay requirements of hard real-time traffic. The idea is to ensure transmission of a real time packet from specified source to destination before expiration delay to deadline.

In view of this, we adopt IntServ principle based on the admission control and the reservation of resources.

In this approach, we implement a two-phase mechanism to be accomplished before the commencement of transmission of real time traffic. One of the phases is responsible for a suitable route selection and the other phase is meant for reservation of resources. The first phase, called as "Real Time Route Discovery (RTRD)", allows to find, if possible, a path between the specified source and the destination, which can satisfy the delay requirements. The second phase, called as "Real Time Route Reply (RTRR)", allows reservation of resources along the path, discovered by the first phase.

4.1 Real Time Route Discovery Phase

In the phase of route discovery, the source is allowed to find a path, if possible, to the required destination, which could satisfy the delay requirements (deadline constraints) for the real time flow. When a node requires to transmit real time data to any other node, at first, it broadcasts a route request packet to all its neighboring nodes with the expiration delay to deadline specified. On receiving the route request packet, each node in the ad hoc network (Activity – 1 in Fig. 3), performs admission control test in order to decide if it can accept a new real time flow or not, as depicted in Fig. 3

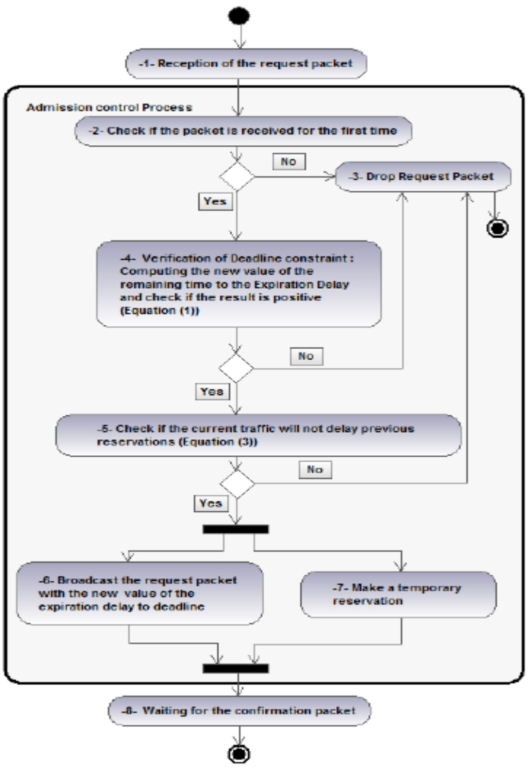


Fig. 3 : Activity diagram of intermediate node behavior in real time route request phase

The purpose behind admission control test is to verify if the network is able to admit the new flow with the specified delay constraints without disrupting the already admitted real time flows in the network. Verification of acceptability of delay constraints, requires two conditions to be checked : one of the conditions is to check if the delay constraints of the new flow can be satisfied, and the other condition is to check if the delay constraints of already admitted real time flows can be maintained in case the new flow is admitted.

Acceptance of first condition (Activity-4 in Fig. 3) refers to the fact that transmission time between source and destination for the new flow is still lower than the expiration delay to deadline. Transmission time includes the time spent at each of the nodes and transmission time between a pair of nodes along the path between source and destination. Thus, each node along the path of the packet, receives the remaining time of expiration delay from the previous node. Consequently, the node subtracts the time taken to transfer the route request packet from the previous node and the processing time of the packet, from this remaining time received. The value so obtained should be positive to validate the deadline on transmission (expiration delay). This relation can be represented as

$$t_i^{k-1} - t_{pk} - t_{tr} > 0 \quad \ldots\ldots\ldots\ldots (1)$$

Where t_i^{k-1} is the remaining time of the expiration delay to deadline, received from (k-1)th node, for i-th traffic, t_{pk} represents the local processing time of the packet at node processing and t_{tr} is the worst-case transmission time between any two neighboring nodes.

In this case, the local processing time is expressed as the sum of the machine processing time and the time spend in the queue.

$$t_p = t_{pm} + t_q \quad \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Where t_{pm} is the machine processing time, and t_q is the time spent by the packet in the queue.

In this case, we assume that node k receives a request packet from node (k-1) with the remaining time to the expiration delay of 10 ms, i.e. $t_i^{k-1} = 10\,ms$. In addition, we presume too, that the local processing time of any packet at node k is 3ms, i.e. t_{pk}=3ms, and the worst-care transmission time between two neighboring nodes is 2ms, i.e. t_{tr} = 2ms. Hence, with validate equation (1) as

$$t_i^{k-1} - t_{pk} - t_{tr} = 10 - 3 - 2 = 5 > 0$$

Since this value is positive, as per condition (1), a packet still has 5ms in reserve to reach the destination before expiration delay to deadline.

Of the two conditions mentioned above, condition (1) verifies if the packet can reach the destination before expiration delay to deadline while condition (2) imposes the new flow, once admitted, not to disrupt the delay requirements of already existing flows (Activity-5 in Fig. 3). To maintain deadlines for existing traffics, we require to save all them with their remaining times, such that an additional traffic must respect their expiration delay to deadlines. For the purpose, we recalculate condition (1), for the accepted flows, as given below. We also take into consideration the changes in case a new flow is admitted.

$$\forall\, j,\ 1 \le j \le adm;\ t_j^{k-1} - t_p - t_{tr} > 0 \ldots .\ (3)$$

Where adm is the number of already admitted flows at the node *
A node can take a decision, with respect to conditions (1) and (3), if it can admit a new real-time flow. If these two conditions are successfully validated for a new real-time flow, then it is admitted, and following it, the node broadcasts a route request packet with the new remaining time to the deadline (Activity – 6 in Fig. 3) and makes a temporary reservation for the flow (Activity – 3 in Fig. 3), and does not send it to any other node, as it cannot satisfy the delay constraints. Hence, it should not belong to the path between the specified source and destination of the real-time flow.

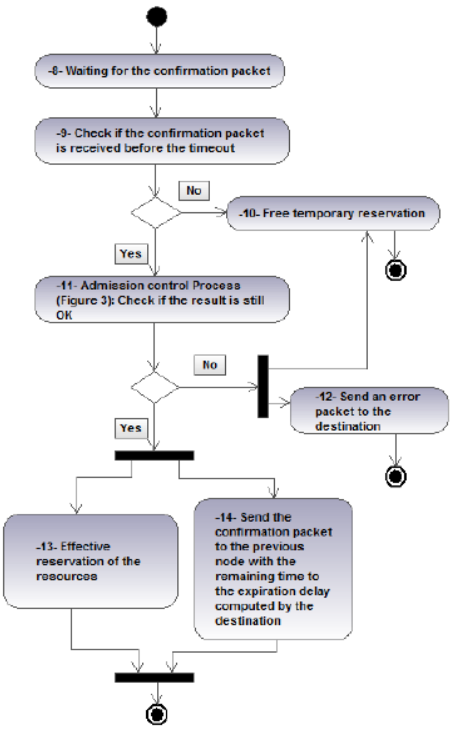


Fig. – 4. Activity diagram of intermediate node behavior in real time route reply phase

4.2.Real Time Route Reply Phase

The role of route reply phase is to reserve a path between source and destination, as specified in the route request packet. Once the delay constraints are satisfied, the destination should communicate the same back to the source and all intermediate nodes along the specified route. Fig. 4 depicts different node activities, implemented in route reply phase.
At the end of the route discovery phase, on receiving the route request packet, the destination performs admission control (Activity – 11 in Fig. 4). If the admission control does not succeed, the destination rejects the route request packet, otherwise, it reserves the route, incorporated in the route request packet, and sends a confirmation packet (route reply packet) along the reversed route, with the final remaining time (Activity – 14 in Fig. 4).
On receiving the route reply packet, each intermediate node along the route of the packet, performs admission control. If admission control fails, the route reply packet is rejected by the node, and an error message is sent to the destination to select another route (Activity-10 in Fig. 4). If admission control succeeds, the node reserves the resources (Activity-13 in Fig. 4), saves the remaining time to deadline and forwards the route reply packet to the next node along the specified route.
If the confirmation (route reply) packet reaches the source, the path is established, following which source starts transmission of data until it has other data to send or the path breaks as a result of node mobility. In such a case, another route discovery phase is initiated to establish a new path, which satisfies the real time constraints imposed by the flow. The nodes of the old path should free their reservations at the same time, which can be achieved by defining a time-out for each reserved flow. Thus, if the traffic does not arrive before the specified time-out, the node will need to free the reservations of resources for the flow.
Achievement of an appropriate path for a real-time traffic is not enough, since availability of a path and respect of real time constraints depend also on mobility of nodes. For the purpose, we address the mobility issue in the following section, in order to provide solutions, even in a mobile environment.

5. Mobility Issue

As ad hoc networks do not support any fixed infrastructure, each node is free to move in the network in an arbitrary manner. Mobility of nodes results in frequent changes in the network topology, which imposes the challenges to maintain real time traffic uninterrupted and delay constraints to be satisfied. In view of this, we analyze different mobility models for ad hoc networks to predict the mobility of nodes in the network such that the real time constraints (deadline) are respected.

5.1. Mobility models for ad hoc networks

Several models for mobility prediction in ad hoc networks have been proposed by authors in [4,16,20,25]. Such models can be classified into two categories : single user mobility models and group mobility models.
Random walk model [26] represents one such single user mobility model, which defines the user movement between two positions with memory less randomly chosen speed and direction. Markovian model also belongs to the category of single user mobility model.
Models like pursue model and column model [19] belong to the category of group mobility models that study the relation between mobile nodes, in disaster recovery and military situations. Reference Point Group Mobility (RPGM) [25], another group mobility model, uses the centre of the mobility group to determine the behavior of the entire group. And, mobility vector model, which offers a flexible mobility framework for hybrid motion patterns [20] represents a group mobility model too.

5.2 Mobility prediction

Mobility issue in ad hoc networks can be effectively resolved by analyzing the history of movement of nodes in the network. Thus, with mobility prediction that causes changes in the network topology, we can predetermine the change in topology, and at the same time, plan a new route discovery at such instants. In between such instants, we presume that the topology remains unchanged. Thus the already reserved resources are freed and a new route discovery is initiated.
In [5,15,26], the authors have attempted to estimate the link availability. A link failure prediction algorithm incorporated in Dynamic Source Routing (DSR) protocol is proposed in [15]. It effectively uses the signal power strength from the received packets to predict the moment of link failure and send a warning message to the source, if the link is to be broken soon. On receiving this message, the source has to perform a pro-active route rebuilt to avoid disconnection. This algorithm implements the following expression :

$$S_r = k \frac{S_t}{(d_o)^4} \quad \ldots\ldots\ldots\ldots\ldots\ldots.. (4)$$

Where S_r is the signal power at receiver, S_t is the signal power at the transmitter, k is any constant between 0 and 1, and d_o is the distance between two nodes. If signal power S_r becomes lower, we can assume that the node will leave its current location and link to this node will be broken.

An iterative algorithm is presented in [6], which continuously predicts link availability between two mobile nodes. In this algorithm, authors estimate the probability (P(d_o,t)) that the link between two mobile nodes will be available uninterrupted in a defined period of time, called epoch time (time between t_0 and t_0+t) with d_0 as the initial distance between these nodes. The calculations are done with assumptions that the initial velocity of the nodes is unknown and the initial relative velocity is known. It should be noted that the current work is used to improve DSR routing protocol and can be useful for many areas in ad hoc networks.
Authors in [5], suggest to predict the stability by characterizing mobility of the nodes, and thus, propose a new scheme for estimation of mobility parameters like relative speed, orientation and epoch time for a real-time application. Estimation of such parameters is achieved from the time-varying inter-node distance information. This principle assumes that every node knows the distances between its one-hop neighbors and the distances separating it to them. These distances are measured periodically using the parameters like either received signal strength (RSSI), time of arrival (ToA), or time-difference-of arrival (TDoA) [5].
Authors in [2], have studied the connectivity of mobile ad hoc networks, and proposed a connectivity model using Markov Jump Theory. Formulation of this connectivity model includes a set of theorems and definitions.
Approaches to mobility prediction can be split into two categories : (i) probabilistic approaches; (ii) deterministic approaches. The probabilistic approaches like Markovian model, are inadequate for hard real time traffic. However, deterministic approaches rely upon prediction of link failure or prediction of node mobility such as a method based on the signal power strength, which is effectively used by us in our current research work.

5.3 Our approach of mobility prediction

Due to mobility of nodes in ad hoc networks, an established route for transmission of real time traffic, may be broken at any point of time. To overcome violations in real-time constraints, we should be able to predict the topology changes and in the worst case, when no more paths can be available for the required real-time traffic. Hence, we perform a new route discovery, incorporating the ideas, presented in the preceding section, and thus, we propose a pre-execution check model for the purpose.
Our pre-execution check model consists of two modules. The first module (Module 1 in Fig. 5) allows to determine possible paths, where as the second module (Module 2 in Fig. 5) checks availability of path and deadline delay satisfaction.

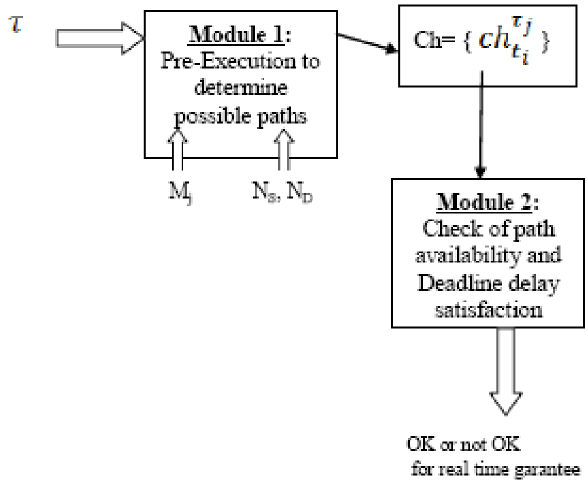


Fig. 5. pre-execution check model to predict real time guarantee

As depicted in Fig. 5, Ch is the set of paths determined from pre-execution module (Module 1), which performs several simulations, to identify nodes behavior. A set of CBR traffic $\tau=\{(d_j,p_j)\}$ is defined, where each traffic is characterized by the expiration delay d_j (deadline) and the inter-arrival period p_j, Ns is the source node and N_d the destination node. M_j represents the movement function of mobile nodes in the ad hoc network.

Path availability and satisfaction of deadline delay are checked by the second module (module 2) in Fig. 5. In this case, we assume that movement of each node in the ad hoc network can be predicted like the predefined movement of a robot in a production unit. Movement of a node is defined by the movement function $M_i(t)$, which refers to the position of a node i at time t. Thus, we obtain the equality $M_i(t)=(x(t),y(t))$, when the node exists at location (x,y) at time t, i.e. x=x(t) and y = y (t). The movement of a node, in such a case, is assumed to occur in a cyclic pattern. Thus, assuming the period of a cycle to be T_i, we obtain $M_i(t+T) = M_i(t)$.

We start with pre-execution to establish paths, for which we adopt property 1, which refers to periodical nature of movement of nodes in the ad hoc network.

Property 1 :

$$M_i(t) \| t \in [0,\infty \Leftrightarrow M_i(t) \| t \in [0,T$$

Where T = PPCM (T_i), i∈ [1,N], with N as number of nodes in the network.

Property 1 allows to check the availability of a selected path only in the period T, as the movement pattern of nodes will be the same in other periods, as depicted in Fig. 6.

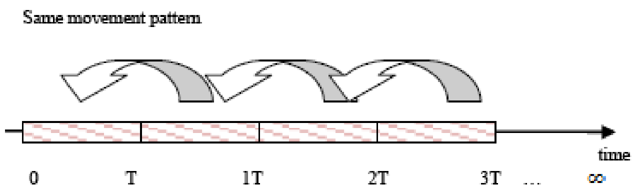


Fig. 6 Periodical characteristic of the nodes movement pattern

When the pre-execution is performed by Module 1 and possible paths are determined, the check model (Module 2) will be initiated to validate different paths with respect to deadline constraints. For this purpose, we need to check two conditions in period T, i.e. condition 1 : path availability; condition 2 : satisfaction of delay constraints.

Condition 1 : Path availability

It allows to determine the duration of availability of the path and checks if data can be transmitted between source and destination along the already selected route.

For this purpose, we assume that the path, chα(t), for the traffic α at time instant t, is available when the neighboring nodes do not leave each other's coverage area. Signal power strength (eq. 4) can be used to predict time of link breakage between two neighboring nodes. Thus, when the signal is detected to be weak, a warning message is sent to the source with information that the link is soon-to-be-broken. On receiving this warning message, the source initiates a route building.

Condition 2 : Satisfaction of delay constraints.

Satisfaction of delay constraints can be verified by checking if the response time R is lower than the expiration delay d (R<d). This can be done from the following expression.

$$\forall t \in [0,T], R\left(ch^j(t)\right) < d_j \quad \text{.......... (5)}$$

Where R is the response time of traffic j, d_j is the expiration delay to deadline and $ch^j(t)$ is the path taken by traffic j at time instant t.

If equation (5) holds true for traffic j, then for each instant of topology changes, the response time for the selected path should be lower than the expiration delay to deadline, and thus, we achieve property 2 as:

$$\forall t \in [0,T], R\left(ch^j(t)\right) < d_j \quad \text{........... (6)}$$

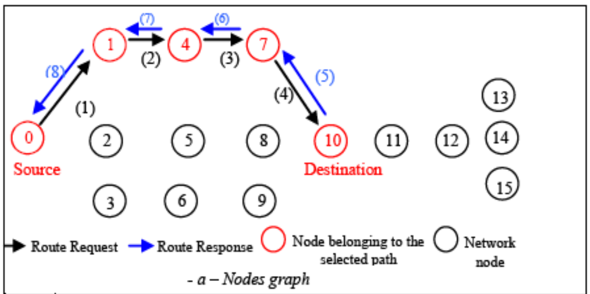
Where {t1,t2,…..tn} is the set of instants of changes in the path of traffic j. It should be noted that the checking instants of real time constraints, expiration delay in particular, are minimized by eq. (6).

6. Evaluation

In our simulations, we have implemented RTD-DSR (Real-Time DSR with Delay constraints) using ns-2, for performance evaluation of our proposed scheme. DSR protocol is introduced to RTD-DSR by incorporating the principle of the proposed admission control based on the expiration delay to deadline. For the purpose, some files of DSR protocol are modified in the source code of ns-2 simulator. Principles of our admission control strategy and reservation of resources are added to file dsragent.cc, which is the DSR agent for handling routing features. To achieve RTD-DSR extensions for routing packet headers, some modifications are brought into the files request-table.cc and hdr_sr.cc.

6.1 Simulation environment

We have carried out our simulations using simulator ns-2, as mentioned before. Our simulation scenario includes 16 mobile nodes in an area of 1300 x 1300m^2. Transmission range of each mobile node is set to 250 meters. Hence, it becomes convenient to represent multi-hop communications without shortcuts.



- a – Nodes graph

```
(1)  Initial value of Expiration delay = 10 in node 0
(2)  The remaining time of the new traffic = 8 in node 1
(3)  The remaining time of the new traffic = 6 in node 4
(4)  The remaining time of the new traffic = 4 in node 7
(5)  I'm the destination and the remaining time = 2 in node 10
(6)  It' s a route reply and the remaining time = 2 in node 7
(7)  It' s a route reply and the remaining time = 2 in node 4
(8)  It' s a route reply and the remaining time = 2 in node 1
```

- b – Execution trace in NS-2

Fig. 7 : Ad hoc network scenario executed in NS2

Our basic simulation network is represented in Fig. 7, where multiple paths can be established between the source (node 0) and the destination (node 10). Here, we chose a matrix sub-network of dimension 3 x 3 (nodes 1,4,7,2,5,8,3,6,9). The simulation parameters are : the workload, the number of traffics, the mobility and the expiration delay to deadline.
As hard real time traffics are used in our simulation network, we attempt to achieve the response time with regard to DSR and our routing protocol RTD-DSR, as results of our simulations. The above two response times are compared to validate the extensions brought into DSR in our proposal.

6.2 Admission Control

While performing admission control, we assume that node 0 wants to send real-time traffic to node 10 with a strict real time delay equal to 10 (Fig. 7). In course of it, we observe the behavior of the nodes on receiving the route request for this flow. As demonstrated in Fig. 7, the value of expiration delay (remaining time to deadline) is decremented by 2, with each node receiving the route request, until it reaches the destination. In this example, the path selected is 0→1→4→7→10. When the route request packet reaches node 10 (destination), the remaining time to deadline is 2, which is still positive, the destination sends a confirmation packet back to source node (node 0), reversing the path. Each inter mediate node including the source, learns from the confirmation packet, that the final value of remaining time to deadline is 2 at the destination. Hence, the previous reservations can still be preserved, without violating the real time constraint, is remaining time to deadline is guaranteed.

6.3 Comparison between RTD-DSR and DSR in a fixed architecture.

We have carried out several simulations observing the delays (Fig.8), to compare transmission delays in RTD-DSR and DSR. We record the transmission delay of packets for a flow using RTD-DSR and DSR, with increasing number of traffics in the network. It can be explicitly observed from Fig. 8, that delay in RTD-DSR, as compared to that in DSR, is convincingly lower and more stable. This stability is resulted from the path reservation for the first traffic. It can also be observed, that for a small number of traffics, both provide the same delay, which is a result of usage of cache by DSR. But, with increasing number of traffics, RTD-DSR clearly demonstrates a better delay and stability.

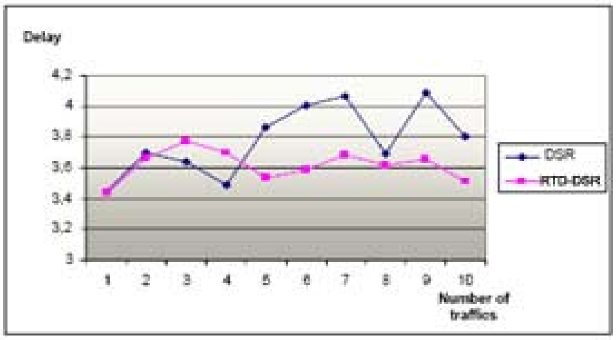


Fig. 8 : Delays depending on the number of traffics

6.4 Comparison of RTD-DSR and DSR with mobile nodes

With moving nodes between the source and destination, the transmission delays, incurred using DSR and RTD-DSR are shown in Fig. 9. We consider some nodes to move in a rectangular manner, eventually going out of each other's transmission range. It can be observed from Fig. 9 that delay of RTD-DSR is still lower than that of DSR, and it is notably stable. Hence, we conclude that our solution is more effective even in a dynamic topology.

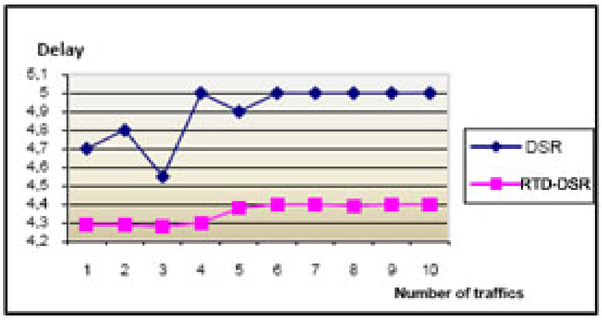


Fig. 9 : Delays depending on the number of traffic in the case of moving nodes

7. Conclusion

In this paper, we present a routing protocol RTD-DSR, which provides QoS solutions to hard real time delay satisfaction in mobile ad hoc networks. It allows the packets of a real time traffic to be transmitted from the source to destination before the expiration delay to deadline, which is achieved with the reservation of a selected path between the source and the destination. The reservations are maintained by an admission control mechanism. We use a pre-execution check model to predict the validity of a selected path and satisfaction of delay constraints. We address the problem of guarantees to delay constraints for hard real time applications with an assumption of a periodical and predictable movement pattern of mobile nodes in an ad hoc network.
Feasibility and performance evaluation of our proposed routing protocol are verified through extensive simulations. Our protocol RTD-DSR offers better delay, which is convincingly lower and more stable, as compared to that in DSR.

8. Future Work

Our protocol RTD-DSR is based on a predictable behavior of mobile nodes in a conservatively dynamic ad hoc network. Extensions to it can be still brought in, with an intention to adapt it in unpredictable and unreliable networks, where nodes can even break down. Devising such a solution requires intensive investigation of features of mobile nodes, their heterogeneous characteristics and many other factors. We expect to achieve the same in our future work.

Reference :

[1] Binod Kumar Pattanayak, Alok Kumar Jagadev, Manoj Kumar Mishra, Dr. Manoj Ranjan Nayak, "Evaluation of A Centralized Bandwidth management Protocol Architecture with Admission Control In Mobile Ad Hoc Networks", International Journal on computer engineering and Information Technology (IJCEIT), Volume 2, No. 2, P. 79-91, Nov 2008-Jan 2009.
[2] Demin Li1, Jie Zhou2 and Jiacun Wang3,"A Markov Jump Theory Based Connectivity Model of Mobile Ad Hoc Networks", 2007 International Symposium on Nonlinear Dynamics (2007 ISND), Journal of Physics: Conference Series 96 (2008) 012012, 2008
[3] B. Hu and H.Gharavi, "DSR-Based Directional Routing Protocol for Ad Hoc Networks", National Institute of Standards and Technology, IEEE Communications Society, IEEE GLOBECOM, 2007.
[4] G. Bos, "QoS support using DiffServ", 6th TSConIT, 2007
[5] Z. Li, L. Sun and Emmanuel C. Ifeachor, "Range-based Mobility Estimations in MANETs with Application to Link vailability Prediction", School of Computing, Communications and Electronics, University of Plymouth, 2007
[6] Z. Li, L. Sun, and E. C. Ifeachor, "Range-based relative velocity estimations for networked mobile devices," Submitted to IEEE Transactions on Vehicular Technology, August 2006.
[7] Min Qin, Roger Zimmermann and Leslie S. Liu, "Supporting Multimedia Streaming Between Mobile Peers with Link Availability Prediction", Department of Computer Science, University of Southern California, Los Angeles, CA 90089, MM"05, November 6–12, 2005, Singapore.
[8] Li Xuefei and L. Cuthbert, "Multipath QoS routing of supporting DiffServ in mobile ad hoc networks", Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, and First ACIS International Workshop on Self-Assembling Wireless Networks. SNPD/SAWN 2005.
[9] A. Proutiere, "QoS in multi-service wireless networks A state of the art", Workpackage: http://www.eurongi.org, Project Acronym: Euro-NGI Project Full Title: Design and Engineering of the Next Generation Internet, towards convergent multi-service networks , August 2004
[10] C. Chaudet, " Autour de la réservation de bande passante dans les réseaux ad hoc", INRIA – Institut National des Sciences Appliquées de Lyon France, septembre 2004
[11] H. Badis, K. Al Agha, "Optimal Path Selection Analysis in Ad Hoc Networks", in: Technical Report; LRI, université de Paris-Sud XI, August 2004.
[12] Y. Yang, "Distributed QoS Guarantees for Realtime Traffic in Ad Hoc Networks", Department of Computer Science, University of Illinois at Urbana-Champaign, October 2004.

[13] A. Laouiti and C. Adjih, “Mesures des performances du protocole OLSR, Hipercom”, in international conference in Sciences of Electronic, Technology of Information and Telecommunications (SETIT), 2003.
[14] X. Hou and D. Tipper, “Impact of Failures on Routing in Mobile Ad Hoc Networks Using DSR”, Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference, Jan., Orlando, 2003.
[15] Gahng-Seop Ahn, Andrew T. Campbell, Andras Veres and Li-Hsiang Sun , “SWAN: Service Differentiation in Stateless”, Wireless Ad Hoc Networks, IEEE, 2002
[16] L. Qin and T. Kunz, “Increasing Packet Delivery Ratio in DSR by Link Prediction”, Department of Systems and Computer Engineering, Carleton University, Ottawa, Ontario, Canada K1S 5B6, IEEE2002
[17] T. Camp, J. Boleng, V. Davies, “A Survey of Mobility Models for Ad Hoc Network Research”, Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, Volume 2, No.5, 2002.
[18] C. Chaudet and I. Guérin-Lassous, “ BRuIT : Bandwidth Reservation under InTerferences influence”, July 2001
[19] D. B. Johnson, D. A. Maltz and J. Broch, “DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks”, 2001.
[20] R. Leung, J. Liu, E. Poon, A. Charles Chan and B. Li, “MPDSR: A QoS-aware Multi-path Dynamic Source Routing Protocol for Wireless Ad-Hoc Networks”, Department of Electrical and Computer Engineering, University of Toronto, 2001
[21] X. Hong, T. Kwon, M. Gerla, D. Gu and G. Pei, “A Mobility Framework for Ad Hoc Wireless Networks”, Proceedings of ACM 2nd International Conference on Mobile Data Management, MDM 2001,Hong Kong, January 2001.
[22] Y. –C Hu and D. B. Johnson, “Implicit Source Routes for On-Demand Ad Hoc Network Routing”, *ACM MobiHoc,* 2001.
[23] Hannan XIAO1, Winston K.G. SEAH2y, Anthony LO2 and Kee Chaing CHUA2, “Flexible Quality of Service Model for Mobile Ad-Hoc Networks”, Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo. 2000 IEEE 51st.
[24] T. LEMLOUMA, Dr. N. BADACHE, “Le Routage dans les Réseaux Mobiles Ad Hoc“, Université des Sciences et de la Technologie Houari Boumèdiene, Institut d'Informatique, Septembre 2000.
[25] S. Basagni, I. Chlamtac, and V. R. Syrotiuk, “Dynamic source routing for ad hoc” networks using the global posit ioning system. In Proceedings of the IEEE Wireless Communications and Networking Conference 1999 (WCNC'99), New Orleans, Louisiana, September 1999.
[26] X. Hong, M. Gerla, G. Pei, C. Chiang, “A Group Mobility Model for Ad Hoc Wireless Networks”, Proceedings of ACM/IEEE International Workshop on Modeling and Simulation or Wireless and Mobile Systems, MSWiM ‟99, Seattle, August 1999.
[27] M. M. Zonoozi and P. Dassanayake, “User mobility modelling and characterization of mobility patterns”, IEEE Journal on Selected Areas in Communications, Volume 15, No.7, September 1997.
[28] D. Ballard and D. Nielson, “A real-time knowledge processing executive for Army rotorcraft applications”, Digital Avionics Systems Conference, 1992.

Alok Kumar Jagadev completed Master in Computer Application from Regional Institute of Technology, Jamshedpur, Bihar state, India in 1992 and M.Tech in computer Science from Utkal University, Bhubaneswar, and Orissa State, India in 200. He is currently continuing Ph.D. in Ad hoc networks in Siksha O Anusandhan University, Bhubaneswar, India. He has published research papers in several reputed national and international journals. He is author of three books in computer science and engineering. He is currently working as an Asst. Prof. a.nd Head in the department of Computer Science & Engineering, ITER, SOAU, and Bhubaneswar, India.

Binod Kumar Pattanayak completed M.S. in Computer Engineering from Kharkov Polytechnical Institute, Kharkov, Ukraine in 1992. He is currently continuing Ph.D. in Ad hoc networks in Siksha O Anusandhan University, Bhubaneswar, India. He has published research papers in several reputed national and international journals. He is currently working as an Asst. Prof. in the department of Computer Science & Engineering, ITER, SOAU, and Bhubaneswar, India.

Manoj Kumar Mishra completed B.E. in Computer Science and Engineering from Marathwada University, Maharashtra State, India in 1992, and M.Tech in computer Science from Utkal University, Bhubaneswar, Orissa State, India in 2001. He is currently continuing Ph.D. in Ad hoc networks in Siksha O Anusandhan University, Bhubaneswar, India. He has published research papers in several reputed national and international journals. He is currently working as an Asst. Prof. and Head in the department of Information Technology, ITER, SOAU, and Bhubaneswar, India.

Manoj Ranjan Nayak completed B.E. in Electrical Engineering from Regional Engineering College, Rourkela, Orissa State, India in 1982, M.Tech in Computer Science and Engineering from Indian Institute of Technology, Kharagpur, West Bengal, India in 1988 and Ph.D. in Computer Science and Engineering from Indian Institute of Technology, Kharagpur, and West Bengal, India in1996. Currently, he is the Chancellor of Siksha O Anusandhan University, Bhubaneswar, India. He has published research papers in more than a hundred reputed national and international journals. More than fifteen research scholars have been awarded the Ph.D. degree under his guidance and several others are still continuing the Ph.D. dissertation work under him.