WCM128

WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2003; **3**:000–000 (DOI: 10.1002/wcm.128)

A resource-efficient QoS routing protocol for mobile ad hoc networks

Swades De*,†

Department of Electrical Engineering
State University of New York at Buffalo
Buffalo
NY 14260
USA

Chunming Qiao

Department of Computer Science and Engineering
State University of New York at Buffalo
Buffalo
NY 14260
USA

Sajal K. Das

Department of Computer Science and Engineering
University of Texas at Arlington
Arlington
TX 76019
USA

Summary

The performance of existing QoS routing protocols is often constrained with high control traffic and database maintenance overhead. We observe that by proper coupling of nodal mobility and location information, better QoS support can be achieved with reduced control traffic and database requirements. In this paper, we investigate the performance of a location-aware QoS routing protocol, called *trigger-based distributed routing* (TDR), for mobile ad hoc networks. In this protocol, the nodal database size is reduced by maintaining only local neighborhood information, and route maintenance control overhead is kept low by maintaining only one route at a time for a session. Distributed rerouting control and directed alternate route discovery help reducing the rerouting control overhead and performing quicker route repair. Moreover, rerouting based on signal degradation history makes it possible to minimize the in-session route failure. Our evaluation shows that the TDR protocol has significantly better QoS support and reduced overhead requirements compared to the existing QoS routing protocols in ad hoc networks. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS
ad hoc networks
on-demand routing
mobility model
performance analysis
real-time quality of service (RT-QoS)
QoS ratio
grade of service (GoS)
global positioning system (GPS)

Q1 *Correspondence to: Swades De, Department of Electrical• Engineering, State University of New York at Buffalo, Buffalo, NY 14260, USA.
†E-mail: swadesd@eng.buffalo.edu
Contract/grant sponsor: Nortel Networks, Richardson, TX.

1. Introduction

Many authors have addressed the routing issues in ad hoc networks from the best-effort service point of view [1–6]. While these approaches attempt to minimize the control and database maintenance overhead in serving the traffic, they do not meet *real-time quality of service* (RT-QoS) criteria, such as bandwidth constraint, end-to-end packet delay, and packet loss. On the other hand, the proposals dealing with QoS provisioning require high control and/or nodal database maintenance overhead [7–10].

We observe that some form of proactive routing scheme has to be adopted to tackle the delay, loss, and bandwidth constraints of real-time applications in ad hoc networks. At the same time, in order to optimize resource utilization, one has to see that the buffer and signaling overhead do not go overboard. To address these issues jointly (i.e. QoS support and resource optimization), one needs to have proper mobility and location information about the nodes. As it has been demonstrated in [2,11], the location information can effectively reduce the route discovery overhead. Likewise, the ability to predict the location of nodes with the knowledge of their mobility would help in efficient discovery of an alternate route.

In this paper, we present a routing algorithm, called *trigger-based distributed routing* (TDR), to deal with link failures (induced by, e.g., nodal mobility) in mobile ad hoc networks. From the network operation point of view, the proposed TDR scheme is a *reactive* algorithm, as the rerouting routine is *triggered* at an active node based on the level and trend of variation of its receive power from the downstream active node. Hence the name 'trigger-based' routing. On the other hand, from the user application point of view, it is a *proactive* algorithm as (ideally) the traffic experiences no break in the logical route during the session, thus making it suitable for dealing with real-time traffic. The routing scheme is also 'distributed' in the sense that any active node participating in a session can make its own routing decision, which helps reducing the nodal computational and database overhead. Our goal is to provide RT-QoS support while keeping the network overhead low. More specifically, to reduce control traffic, we propose to maintain only the active routes and exploit the location information of the destination to *selectively forward* alternate route queries when a link failure is imminent. Our evaluation shows that the proposed TDR protocol provides better QoS support with lower control overhead in comparison with the schemes in [7,9], which operate without link failure prediction capability. The TDR scheme provides QoS support comparable to Flow Oriented Routing Protocol (FORP) [10] while incurring substantially lower control overhead.

The rest of this paper is organized as follows. Related previous work is surveyed in Section 2. The TDR protocol details are provided in Section 2. Mobility models are discussed in Section 4, based on which route lifetime and associated control overhead can be quantified. Section 5 provides analysis of the protocol performance in terms of the reduction in control overhead due to selective route search. Section 6 presents simulation-based performance evaluation and comparison results. Section 7 concludes the paper.

2. Previous Work

A lot of work has been reported on routing protocols for mobile ad hoc networks. While the *reactive* (or on-demand) algorithms, such as DSR [1], TORA [3], ABR [6], ZRP [4], AODV [5], and •GPSR [2], operate with limited control and database maintenance overhead, they are suitable only for delay-tolerant applications. On the other hand, the *proactive* (or table-driven) approaches, such as DSDV• [8], WRP [12], GSR [13], and DREAM [14], attempt to minimize the route disruption time (hence packet loss), but are encumbered with high control and database maintenance overhead. Q2 Q3

Recently, some QoS-capable protocols have been reported. For example, a protocol with QoS extension to AODV [9], called *E-AODV*, addresses the bandwidth and delay guarantee requirements. Route discovery in this protocol is broadcast-based. The reactive nature of the protocol does not help minimize the service disruptions due to nodal mobility. An in-band signaling approach for supporting QoS, called *INSIGNIA*, is presented in [15]. A route is discovered by the inflow packets and is maintained at the active nodes by velocity-dependent 'soft-state' tags. Since the nodes are not responsible for maintaining the flow state information, in case of route failure, duplicate and out-of-order packet delivery can still occur. The Distributed Quality-of-Service Routing (which we call DQoSR) scheme, proposed in [7] for meeting bandwidth and/or delay constraints, requires that a number of secondary routes be maintained in addition to the primary (currently in use) route to the destination. The network state information at each node, obtained via periodic beaconing, enables finding the routes to the destination by a limited number of 'tickets'. But this costs extra database and bandwidth. In particular, the nodal database will grow at the same rate with network size as in DSDV. In FORP [10], the flow states are maintained for QoS support, aided by the predicted link expiration times. Rerouting is controlled by the destination node, and route discovery at any phase is broadcast-based. QoS extension of DSR [16] suggests flow state maintenance to minimize the route disruption for a session. Implementation of proactive routing on top of DSR and AODV for providing QoS support is reported in [17]. The routing scheme in these approaches [16,18] is source-controlled and route discovery is broadcast-based.

3. Trigger-based Distributed Routing

The proposed TDR protocol is designed to support RT-QoS-aware applications. The scheme makes use of on-demand route discovery, as in DSR, AODV, ABR, TORA, and GPSR, to reduce the control overhead. To maintain the RT-QoS constraints, the flow state for each session is maintained, as in FORP [10], but in a distributed fashion at the active nodes. In case of imminent link failure in the active route, alternate route searching overhead is kept low by localizing the alternate route queries to within certain neighbors of the nodes along the source-to-destination active route. For cost efficiency (quicker search and reduced control overhead), rerouting is attempted from the location of an imminent link failure, which we denote as *intermediate node initiated rerouting* (INIR). If INIR fails, to keep the flow state disruption at a minimum, rerouting is attempted from the source node, which is termed as *source-initiated rerouting* (SIRR)‡. The TDR scheme keeps the size of the nodal database small, irrespective of the network size, by maintaining only the local neighborhood information. In addition, an activity-based database is maintained at each node whose size is limited by its maximum data-handling capacity and interference from the other nearby nodes. The protocol details are described below.

3.1. Database Management

All nodes in the network maintain the local neighborhood information. In addition, for an ongoing session, depending on its activity, a node maintains one of the following three information bases: source database, intermediate node database, and destination database. Note that we have shown only the message fields specific to the TDR protocol. To cope with the wireless channel–dependent errors, the messages can be protected with suitable forward error correcting code.

3.1.1. Local neighborhood database

A node can be in either of the two states—idle (when it is not involved in any session) and active (when it participates in a session). In any state (idle or active), a node n periodically broadcasts beacons containing its location and mobility information to its local neighbors. It also listens to the beacons and maintains a local neighborhood database denoted as link table, LT_n, as shown in Table I. The nodes keep the neighborhood information up-to-date by adjusting the beaconing frequency, depending on the relative mobility of the neighbors. The location information of a node is assumed to be• available from the global Q4 positioning system (GPS), in outdoor environment, or from acoustic range finding devices [20], in indoor environment.

Note that unlike TDR (as well as FORP, GPSR, and E-AODV), which maintains only the local neighborhood database, DQoSR maintains the global information (delay, bandwidth, and cost to all possible destinations) at each node. Assuming the size of the database for each nodal information to be the same in both cases, in an N-node network with n_g neighbors on average, DQoSR would need to maintain a nodal database that is approximately (N/n_g) times larger than that of TDR. This also indicates that for the same network density, the nodal database size in DQoSR grows linearly with network size.

3.1.2. Activity-based information

Besides the neighborhood information, if a node actively participates in a session as the source (S), the destination (D), or an intermediate node (IN), a corresponding table called a source table ST_n, a

Table I. Link table information fields at node n for the ith neighbor.

LT_n	Field description
P_i	Receive power level
X_i, Y_i	Current (X, Y) coordinate
Vel_i, Dir_i	Velocity, direction of motion

‡ It may be noted that INIR followed by SIRR (i.e. INIR + SIRR) scheme is akin to the crank-back route searching approach in ATM - PNNI routing [19].

Table II. Activity-based information fields in different databases at node n.

$ST_{n\cdot}$	$IT_{n\cdot}$	$DT_{n\cdot}$	Field description
Session_ID	Session_ID	Session_ID	Session ID
S_ID	S_ID	S_ID	Source ID
D_ID	D_ID	D_ID	Destination ID
······	S_loc	S_loc	Source location (X, Y)
Max_BW	Max_BW	Max_BW	Maximum bandwidth demand
Max_Del	Max_Del	Max_Del	Maximum acceptable delay
D_loc	D_loc	······	Destination location (X, Y)
N_ID	N_ID	······	Next node ID (towards D)
······	P_ID	P_ID	Previous node ID (towards S)
······	Dist	Dist	Distance from S (hop count)
Nod_actv	Nod_actv	Nod_actv	Activity flag (0 or 1)

destination table DT_n, or an IN table IT_n is maintained. The fields in the three types of databases are shown in Table II, where the first three fields (*Session_ID*, *S_ID*, and *D_ID*) uniquely identify a session. The other fields are maintained for routing information exchange, to be explained in •Section 3.2.

At any time instant, a node n may require to maintain some or all of the tables ST_n, IT_n, and DT_n simultaneously for different ongoing sessions. Contrary to the wireline networks, in which link capacities (bandwidth) are independent of a node's connectivity, in wireless networks a node's data-handling capacity is limited by the node's allocation of bandwidth. For example, if the MAC• layer protocol is CDMA-based, then a node's maximum data rate is limited by multiuser interference and the number of available orthogonal codes (if multicoding scheme is used). If the MAC protocol is TDMA-based, then it is limited by the available time slots, frequency spectrum, and cochannel interference. Accordingly, each node n (idle or active) also maintains an updated residual bandwidth ($Resi\_BW_n$), which indicates its ability to participate in a session. Since the maximum bandwidth resource is limited, the number of sessions that a node can participate in is also limited, irrespective of the network density and size. Therefore, the size of the activity-based database is also limited. The activity-based database is soft-state-maintained and requires to be refreshed by in-session data packets. At any time, if at a node (n) the soft-state timer for a session expires (e.g. as a result of unforeseen route failure), the corresponding nodal database is purged and the $Resi\_BW_n$ is refreshed.

3.2. Control Traffic Management

To maintain updated routing information (activity-based database) at the nodes, certain information exchange among the active nodes is necessary. The required messages to be exchanged for initiating, maintaining, and terminating a real-time session are discussed below.

3.2.1. Initial route discovery

To reduce control traffic, TDR uses two-dimensional location information. However, since an idle node keeps only the local neighborhood information, while initiating a session the source node may not have any clue about the location of the destination unless it is a local neighbor, or its location information is cached at the source node among its recently concluded sessions. If the information is available in the source cache, route discovery is performed via selective forwarding, where the query packet at each node is forwarded to a limited number of preferred neighbors, and this process is repeated until the query reaches the destination. Since the destination's location information in the cache may not be up-to-date (i.e. may be imprecise), the diameter (measured by the number of route request forwarding nodes) of selective broadcast should be larger than that of the alternate route search (to be discussed in Section 3.2.3). In case of no prior knowledge about the destination, the source initiates flooding-based initial route discovery. To ensure stability of routes and to reduce control overhead, only the selected neighbors from where the receive power are more than a threshold level (P_{th1}) are considered for a possible link.

The fields in the initial route discovery control packet are shown in Figure 1. Description of the fields can be found in Table II. Each source provides its own *Session_ID*. To reduce the field size, the lowest possible sequence number is picked up, excluding the IDs for the ongoing sessions originated from that node, as a new *Session_ID*.

The source (S) checks if it has enough residual bandwidth ($Resi\_BW_S$) to satisfy the maximum bandwidth§ requirement (*Max_BW*) for the session. If the

Session_ID	S_ID	D_ID	S_loc	N_ID	Dist	Max_BW	Max_del

Fig. 1. Session initiation route discovery packet structure.

§ This is to ensure full QoS support whenever a flow path is ensured. One could instead consider minimum bandwidth criteria for a flexible QoS support.

demand can be met at S, the required bandwidth is temporarily reserved for a certain lifetime within which it expects to receive the acknowledgment from the destination. The source table ST_S is built with the *Nod_actv* flag still set to '0' (i.e. idle), and the route discovery procedure is initiated. To find a valid route to the destination, a modified breadth-first search algorithm is applied abiding by the following rules:

- Upon receiving the first discovery packet for a session, the IN increments the *Dist* tag by 1 and checks for its residual bandwidth ($Resi\_BW_{IN}$). If it can meet the maximum bandwidth demand, and the updated *Dist* tag is less than *Max_del* (measured as hop count), the required bandwidth is temporarily reserved, the activity table IT_{IN} is built with the *Nod_actv* flag '0', and the packet is forwarded to its downstream neighbors with the updated *N_ID* field. If either or both the *Max_BW* and *Max_del* criteria cannot be satisfied, the discovery packet is simply dropped.
- To ensure loop-free routing, intermediate nodes accept the route discovery packet only once (the one with the minimum *Dist* tag) for a particular session.
- Upon reception of the first discovery packet, if the destination satisfies the *Max_del* requirement (after incrementing the *Dist* tag) and has at least *Max_BW* available, the discovery packet and the corresponding route are accepted. This also ensures the shortest route from the source satisfying the bandwidth and delay criteria.

The above rules ensure that the decided route is the shortest one in the current network condition, which may not be necessarily the shortest in number of hops.

The concept of temporary reservation of bandwidth in the route discovery phase in TDR is similar to that in resource reservation protocol (RSVP) [21], but differs in implementation. More specifically, unlike in RSVP, to minimize the resource holding, the reservation time in TDR is varied depending on the node's location, which is approximately known from the *Max_del* tag and the updated *Dist* tag in the discovery packet. The closer the *Dist* tag to the *Max_del* value, the lesser the reservation time. Let T_d be the current *Dist* tag value at a node and T_M be the *Max_del* requirement for the session. Then the maximum temporary bandwidth reservation time at that node is $2(T_M - T_d)\tau_h$, where τ_h is the maximum time required for a discovery packet to proceed from one node to another, which includes packet processing and propagation time.

3.2.2. Route/reroute acknowledgment

Once a route is accepted, the destination node builds the DT_D table with the *Nod_actv* flag set to '1' (i.e. active) and initiates a route acknowledgment (ACK) message toward the source along the selected route. On receiving the ACK packet, all intermediate nodes and the source node update the fields in their respective *IT* and *ST* tables (i.e. set their *Nod_actv* flags to '1') and refresh their *Resi_BW* status. Once the logical flow path is set up, the packet transmission for the session can follow immediately. The fields in a route/reroute acknowledgment packet are shown in Figure 2.

Besides acknowledging the route/reroute queries, the destination node also sends its location update to the active nodes via the ACK packet whenever there is appreciable change in its location. This reduces the chance of using stale location information for rerouting purposes.

3.2.3. Alternate route discovery

Rerouting a QoS session is necessary when an active node notifies its imminent shutdown state or its receive power from its local active neighbor reduces beyond a certain critical limit. In any case, the upstream active node (closer to the source) initiates the rerouting process. We denote this as *link degradation triggered rerouting*¶.

The rerouting process can be either source-initiated, called SIRR, or intermediate node–initiated, called INIR. An intermediate active node (IN) monitors its downstream receive power level. In SIRR, when the receive power level at an IN decreases to the threshold P_{th2} (see Figure 3), the IN sends a rerouting indication via a 'status query' packet to the source node with the call identification fields (*Session_ID*, *S_ID*, *D_ID*) and the *RR_stat* flag set to '1'. Henceforth, the source takes control of the rerouting process. This rerouting approach is similar to that

Session_ID	S_ID	D_ID	S_ID/IN_ID	D_loc	P_ID	Dist	Max_BW	Max_del

Fig. 2. Route/reroute acknowledgment packet structure.

¶ Other than for increased internodal distance, link degradation can also occur owing to channel-fading effects caused by the inherent nature of the wireless medium. The slow fading problem can be tackled by this scheme. For fast fading, conventional protection mechanism at the data link layer has to be incorporated.

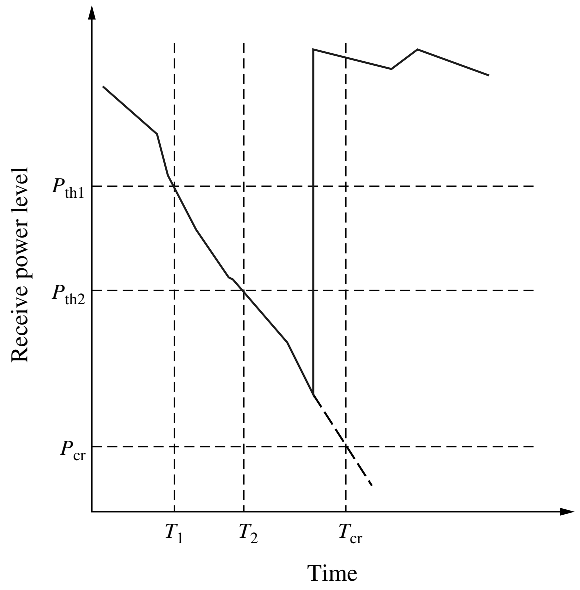


Fig. 3. A pictorial representation of the rerouting process.

in [17], but differs in selective forwarding of route requests.

On the other hand, in INIR, when the downstream receive power level at an IN falls below a threshold P_{th1} with a negative rate of change, it initiates a 'status query' packet toward the source with appropriate call identification fields, filling the *QN_ID* (querying node ID) and *N_ID* fields with its own ID, the *P_ID* field with its previous node in the active route, and with the *RR_stat* flag set to '0'. If any upstream node is in the rerouting process, upon reception of the 'status query' packet it sets the *RR_stat* flag to '1' and returns the packet (as a 'status reply') to the querying node (*QN_ID*). On arrival at the source, the 'status query' packet is discarded (implying that the querying node can initiate the rerouting process). If the query-initiating node receives no reply before its power level from the downstream node goes below the second threshold, P_{th2}, and further tends to decrease, it triggers the alternate route discovery process. Otherwise, it relinquishes the control of rerouting. This query/reply process eliminates the chance of duplicate alternate route discovery for a session. If the downstream receive power at any active intermediate node goes below a critical limit P_{cr}, the source-destination route gets disrupted until the source is able to set up an alternate route. As in handoff in cellular systems [22], selection of thresholds P_{th1} and P_{th2} have to be judicious so that unnecessary rerouting is avoided and at the same time a successful rerouting is done in case of a genuine link failure. The status query/reply packet structure is shown in Figure 4.

An example of rerouting due to link degradation in the active route is shown in Figure 5, where it depicts the INIR. The size of a node indicates the level of bandwidth usage at that node and the thickness of a link denotes the amount of traffic carried along that link (possibly belong to multiple sessions). Since TDR has distributed control, it inherently adopts the INIR scheme. If INIR fails, to avoid/minimize route disruption SIRR is also attempted. It may be noted here that the preemptive routing in [17] follows SIRR. Owing to this, and also since it does not use the location information, the routing/rerouting control overhead in this approach is expected to be more control overhead–intensive.

In either rerouting approach (SIRR, INIR), the alternate route discovery packet structure as shown in Figure 6 can be used. The process is similar to the

Session_ID	S_ID	D_ID	QN_ID	P_ID	N_ID	RR_stat

Fig. 4. Route status query/reply control packet structure.

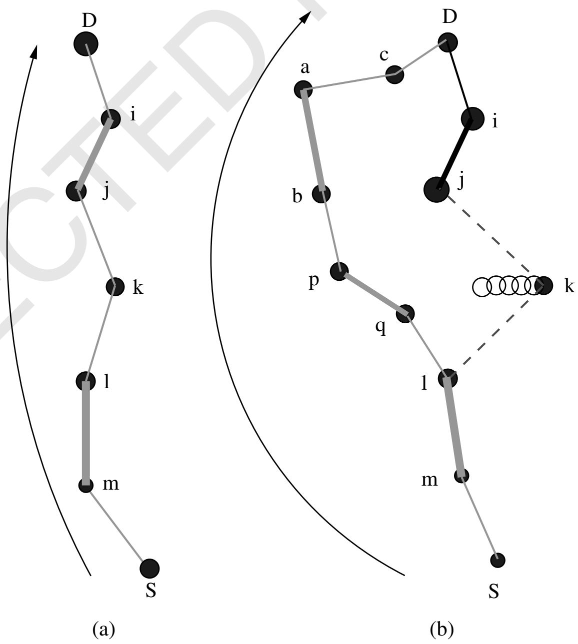


Fig. 5. An example of link degradation–based rerouting. The thickness of a link/node denotes the relative amount of traffic handled by it.

Session_ID	S_ID	D_ID	S_ID/IN_ID	D_loc	N_ID	Dist	Max_BW	Max_del

Fig. 6. Alternate route discovery packet structure.

initial route discovery, except that the packet forwarding from a node in this case is done more selectively. Particularly, the rerouting process takes advantage of location information of the local neighbors and the approximate location of the destination, and forwards the rerouting requests to only selected neighbors closest to the destination satisfying the delay and bandwidth constraints.

The members of this selective broadcast group can change because of nodal mobility, network density, and traffic intensity. For highly mobile scenarios, link degradation occurs fast. In such cases, as well as owing to outdated location information, the membership count can be increased to ensure an alternate route at the appropriate time.

Note that location-aided routing• (LAR) [11] uses the location information in a different way. On the basis of the destination's approximate location, it defines a conical region from the source, and all nodes within the cone are responsible in forwarding the route query. In case of route search failure, the cone angle is expanded. Clearly, depending on nodal density, latency in route search in this approach can vary widely. Route searching control overhead in LAR is also a function of nodal density. In contrast, our local neighborhood information based selective forwarding approach does not have this dependency.

The approach of selective forwarding of route/reroute query in TDR is similar to the geographic forwarding in GPSR [2]. In GPSR, on the basis of the local neighbors' location information, the *actual data packet* is forwarded to the downstream neighbor that is closest to the destination. If at any point no closer neighbor than itself to the destination is found, then the packet is forwarded along the perimeter of the 'void'—called *perimeter forwarding*. The distinct feature in TDR, however, is that it selectively forwards the query to more than one downstream neighbor. As will be shown in Sections 5 and 6, with an optimum number of forwarding nodes at each node along the route, TDR avoids encountering a 'void', and at the same time significantly reduces the number of the• control packet exchanges (when compared with the flooding-based approaches).

3.2.4. Route deactivation

When a session is either finished, terminated, or rerouted, the old route has to be released. In the case of a session completion or termination, the source

Session_ID	S_ID	D_ID	S_ID/IN_ID	N_ID

Fig. 7. Route deactivation packet structure.

node purges its corresponding *ST* table and sends a route deactivation packet through the old route to the destination. The packet structure is shown in Figure 7. Upon receiving a route deactivation packet, a node updates its *Resi_BW* (by releasing the reserved bandwidth) and purges the activity database (*IT* or *DT*) for that session. No explicit deactivation packet is sent in case of rerouting, as the new route could consist of some old active nodes. The departed nodes refresh their activity databases and residual bandwidths after a certain fixed 'soft-state' interval (as in E-AODV [9] or RSVP [21]). Also, if for some reason (e.g. fast link failure) an old route could not be released, the associated nodes refresh their *Resi_BW* and clear their respective activity-based tables after a fixed 'soft-state' interval.

Before we proceed to evaluate the TDR protocol performance, a few comments about the related approaches are in order. The rerouting approach in TDR has some similarities with ABR [6]. Particularly, in both TDR and ABR, routes are constructed as required, and only one route per session is maintained at a time. Route selection in both cases takes care of longevity of links and nodal traffic conditions. Also, to reduce control overhead and searching time, both TDR and ABR attempt rerouting traffic from the point of route failure. The distinct features on TDR with respect to ABR are as follows: (i) Rerouting in ABR is attempted only when a failure is detected, whereas in TDR it is decided prior to the actual link failure, on the health of the immediate downstream link at an active node. (ii) Unlike in ABR, route status query in TDR helps avoid simultaneously initiated alternate route search processes by more than one active node, which in turn reduces rerouting control message exchange in the network. (iii) To reduce bandwidth and energy resource requirements, TDR exploits approximate location information of nodes and restricts the alternate route search query to a limited number of nodes. On the other hand, alternate route query in ABR is always broadcast-based. (Note that in localized query (LQ[H]) approach in ABR, it limits the broadcast range to a certain number of hops, H.)

With the TDR protocol details discussed above, we next proceed to analyze and evaluate the performance of the protocol.

4. Mobility Modeling

In this section, we develop an analytical framework for determining the average lifetime of a route, which is useful for estimating the rerouting control overhead associated with it (in Section 5) and for evaluating the proposed TDR protocol performance (in Section 6).

For obtaining the link lifetime, the approach in [23] for wireless cellular networks is followed, and the two-body-mobility in wireless ad hoc networks is reduced to a one-body-mobility problem by introducing relative position and motion [24]. Since the internodal communication range in an ad hoc network is expected to be small (on the order of microcellular/picocellular BS to MH communication distance), we proceed with the assumption, based on the observations in [25,26] on 'well behaved' users' mobility patterns, that during the lifetime of a given active link, the relatively mobile node moves along a specific direction with a constant velocity. The velocity distribution of different nodes at different time intervals conforms to a given velocity profile.

We derive the route lifetime assuming two different velocity profiles, namely, uniformly distributed and Rayleigh-distributed, as we anticipate that these two profiles would broadly capture two groups of users' mobility pattern. Specifically, a coherent group of users (e.g. military/rescue personnel) have nearly the same velocity that may be represented by Rayleigh-distributed profile. On the other hand, a broad class of users' (e.g. civilians) velocities can be better represented by uniform distribution. Note that in on-demand multipath routing analysis [18], without considering the actual mobility profile, the link lifetime is assumed exponentially distributed. However, as will be observed in the following text, link lifetimes for both the velocity profiles (uniform and Rayleigh) are quite different from those with• exponential nature. Q9

4.1. Uniformly Distributed Velocity Profile

In this model, each mobile node is assumed to have a uniformly distributed velocity between 0 and V_m, and a uniformly distributed direction between 0 and 2π. The mobility of a node is characterized by $f_V(v)$ and $f_\Theta(\theta)$, denoting respectively the velocity pdf (probability density function) and the direction pdf. The two pdfs are defined as follows:

$$f_V(v) = \begin{cases} \frac{1}{V_m}, & 0 \le v \le V_m \\ 0, & \text{otherwise} \end{cases}$$

and

$$f_\Theta(\theta) = \begin{cases} \frac{1}{2\pi}, & 0 \le \theta \le 2\pi \\ 0, & \text{otherwise} \end{cases}$$

With the introduction of relative mobility, the relative velocity (v_e) of an active node will be a uniformly distributed random variable (RV) between 0 and $2V_m$, while the direction RV remains the same. The other neighboring active node of a link is now relatively static at a point. Thus, the new effective pdfs of the relatively mobile node are given by

$$f_{V_e}(v_e) = \begin{cases} \frac{1}{2V_m}, & 0 \le v_e \le 2V_m \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

and

$$f_{\Theta_e}(\theta_e) = \begin{cases} \frac{1}{2\pi}, & 0 \le \theta_e \le 2\pi \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

The pdf of distance Z traversed (refer to Figure 8) by the relatively mobile active node within the range of the relatively static neighbor can be found by using the standard methods [23]:

$$f_Z(z) = \begin{cases} \frac{2}{\pi R^2}\sqrt{R^2 - \left(\frac{z}{2}\right)^2}, & 0 \le z \le 2R \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where R is the range of circular coverage of a mobile node, assumed equal for all nodes.

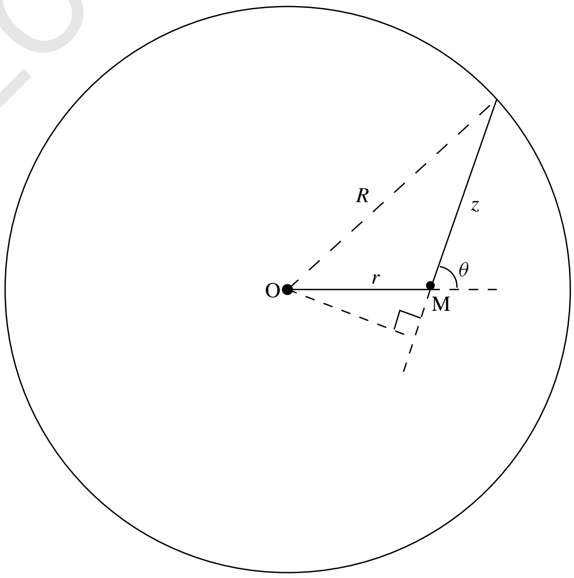


Fig. 8. Distance traversed by the mobile node (M) within the range of relatively static active neighbor (O).

The corresponding pdf of link lifetime ($T_L = Z/V_e$) is

$$f_{T_L}(t) = \int_{-\infty}^{\infty} |v_e| f_Z(tv_e) f_{V_e}(v_e)\, \mathrm{d}v_e \qquad (4)$$

Substitution for $f_Z(\cdot)$ and $f_{V_e}(\cdot)$ from Equations (1 and 3) into Equation (4) gives

$$f_{T_L}(t) = \begin{cases} \dfrac{4R}{3\pi V_m t^2}\left[1 - \left\{1 - \left(\dfrac{V_m t}{R}\right)^2\right\}^{\frac{3}{2}}\right], & 0 \le t \le \dfrac{R}{V_m} \\ \dfrac{4R}{3\pi V_m t^2}, & t > \dfrac{R}{V_m} \end{cases} \qquad (5)$$

whose corresponding CDF (cumulative distribution function) is given by

$$F_{T_L}(t) = \begin{cases} \dfrac{2}{\pi}\sin^{-1}\left(\dfrac{V_m t}{R}\right) - \dfrac{4}{3\pi}\tan\left[\dfrac{1}{2}\sin^{-1}\left(\dfrac{V_m t}{R}\right)\right] \\ \quad + \dfrac{1}{3\pi}\sin\left[2\sin^{-1}\left(\dfrac{V_m t}{R}\right)\right], & 0 \le t \le \dfrac{R}{V_m} \\ 1 - \dfrac{4R}{3\pi V_m t}, & t > \dfrac{R}{V_m} \end{cases} \qquad (6)$$

In a K-hop source-to-destination route in a multihop wireless network with independent link failures, the link lifetime RVs are independent. Denoting the link lifetime RVs as $T_{L_1}, T_{L_2}, \ldots, T_{L_K}$, the route lifetime ($T_R$) is expressed as

$$T_R = \min\left(T_{L_1},\ T_{L_2},\ \ldots,\ T_{L_K}\right) \qquad (7)$$

Q10 For i.i.d.• RVs, the route lifetime pdf is obtained as

$$f_{T_R}(t) = K f_{T_L}(t)\left(1 - F_{T_L}(t)\right)^{K-1} \qquad (8)$$

where $f_{T_L}(t)$ and $F_{T_L}(t)$ are given by Equations (5 and 6), respectively.

The expected K-hop route lifetime,

$$\overline{T}_R = \int_0^{\infty} t f_{T_R}(t)\, \mathrm{d}t \qquad (9)$$

is obtained from Equation (8).

4.2. Rayleigh-distributed Velocity Profile

The velocity of a mobile node with parameter σ in this model is characterized by

$$f_V(v) = \begin{cases} \dfrac{v}{\sigma^2}\mathrm{e}^{-v^2/\sigma^2}, & v \ge 0 \\ 0, & \text{otherwise} \end{cases}$$

with uniformly distributed mobility direction between 0 and 2π.

For two mobile nodes with Rayleigh-distributed velocity profile with respective parameters σ_1 and σ_2, the relative velocity of one with respect to the other relatively static node is Rayleigh-distributed with effective parameter $\sigma_e = \sqrt{\sigma_1^2 + \sigma_2^2}$. The effective mobility profile of the relatively mobile active neighbor is therefore characterized by

$$f_{V_e}(v_e) = \begin{cases} \dfrac{v_e}{\sigma_e^2}\mathrm{e}^{-v_e^2/\sigma_e^2}, & v_e \ge 0 \\ 0, & \text{otherwise} \end{cases} \qquad (10)$$

with the same $f_{\Theta_e}(\theta_e)$ as given by Equation (2).

From Equations (3, 4, and 10), the pdf of link lifetime ($T_L = Z/V_e$) is obtained as

$$f_{T_L}(t) = \begin{cases} \dfrac{2}{t}\,\mathrm{e}^{-R^2/\sigma_e^2 t^2} I_1\left(\dfrac{R^2}{\sigma_e^2 t^2}\right), & t \ge 0 \\ 0, & \text{otherwise} \end{cases} \qquad (11)$$

where R is the range of a mobile node and $I_\nu(x)$ is the modified Bessel function of the first kind of order ν with parameter x.

Hence, the corresponding CDF is

$$F_{T_L}(t) = \begin{cases} e^{-t'}\left[I_0\left(t'\right) + I_1\left(t'\right)\right], & t \ge 0 \\ 0, & \text{otherwise} \end{cases} \qquad (12)$$

where $t' = R^2/\sigma_e^2 t^2$.

Assuming the link lifetime RVs to be i.i.d., the pdf of K-hop route lifetime (Expression 7) is given by Equation (8), where f_{T_L} and F_{T_L} are given by Equations (11 and 12) respectively. Hence, by Equation 9, the expected K-hop route lifetime can be obtained as

$$\overline{T}_R = \int_0^{\infty} \left\{1 - \mathrm{e}^{-t'}\left[I_0\left(t'\right) + I_1\left(t'\right)\right]\right\}^K \mathrm{d}t \qquad (13)$$

where $t' = R^2/\sigma_e^2 t^2$.

Numerical results for expected route lifetime, corresponding effective control overhead (ECOH), and verification with simulation results are presented in Section 6.

5. Routing Performance Analysis

In this section, we quantify the routing performance and resource gain associated with TDR. Note that although it is intuitively obvious that the wider the route query zone (in terms of the number of query forwarding nodes from an upstream node), the higher will be the probability of a successful query search, we are also interested in minimizing the rerouting overhead. In this section, we will analyze the optimum number of query forwarding nodes required to ensure an alternate route.

5.1. Average Number of Neighbors

First, we obtain the average number of neighbors surrounding a node in the mobility space. Considering N nodes are distributed uniformly over a mobility space of area A, the approximate number of neighbors (n_g) of a node is given by

$$n_g \approx \sum_{i=1}^{N-1} i \times \Pr\{\text{the node has } i \text{ neighbors}\}$$
$$= \sum_{i=1}^{N-1} i \times C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \quad (14)$$

where a is the coverage area of a mobile node (considered equal for all nodes).

For simplicity, Equation (14) does not consider the 'boundary effects' where the nodes near the boundaries will have lesser region covered within the rectangle and hence there would be less than the predicted number of nodes around them. However, the error in the estimate (without considering the 'boundary effects') and in the subsequent analysis would be negligibly small for smaller a, larger N, and larger A. An approach to an accurate estimate of the average neighbor count is provided in Appendix I. The 'boundary effect' approximation error is shown in Table III.

5.2. Probability of Successful Alternate Route Search

To obtain a successful alternate route search probability in location-based directed query and to determine the optimum value of the maximum number of query forwarding nodes, we begin with the route search failure probability for the case of only one forwarding neighbor. Subsequently, we obtain the failure probability for more relaxed cases, with more than one forwarding neighbor.

Table III. 'Boundary effect' on average number of neighbors around a node. $N = 150$, $R = 300$ m ($a = \pi R^2$).

A (m^2)	Approximate n_g [Equation (14)]	Accurate n_g [Appendix I]
2000 × 1500	14.0	12.0
2000 × 2000	10.5	9.2
2500 × 2000	8.4	7.5

In selecting one or more forwarding neighbors of a node out of all its neighbors, it is assumed that the ones closest to the destination (based on relative location information) qualify first. Assuming that a node can serve one call at a time, a local neighbor is available for routing the query packet if it is not already acting as either a source, or an intermediate node, or a destination. For Poisson call arrival process at a node with rate λ and the average call holding time $\overline{x}$, the probability that a node is busy as a source, p_s, is

$$p_s = \lambda\overline{x}$$

Considering equiprobable source–destination pairs, a node can act as a destination with probability $1/(N-1)$. There are $N-1$ such potential nodes that could choose it as a destination. Therefore, the probability that a node is busy as a destination, p_d, is

$$p_d = \lambda\overline{x}$$

If the average route length is h-hop long, there would be on average $(h-1)$ nodes acting as intermediate nodes (i.e. routers) for a call. Hence, the probability that a node is busy as a router, p_r, is

$$p_r = (h-1)\lambda\overline{x}$$

Summing up all these, the probability that a node is busy, p_b, is given by

$$p_b = (h+1)\lambda\overline{x} \quad (15)$$

Assuming that each session takes a fixed (same) amount of bandwidth, if a node can support c such real-time sessions simultaneously, then Equation (15) will be modified as $p_b = (h+1)\left(\frac{\lambda\overline{x}}{c}\right)$. In any case, as a stability criteria the values of λ, h, c, and $\overline{x}$ should be able to satisfy the condition $p_b < 1$. For example, given an average call holding time ($\overline{x}$), if the average hop length (h) is longer, the call arrival rate (λ) has to be lower and/or the number of simultaneously supported calls at a node (c) has to be higher.

Case 1: *Only one forwarding neighbor:*

Irrespective of the number of forwarding neighbors, a query packet forwarding at the source node succeeds if at least one of its local neighbors is available (i.e. can accommodate the call) at that instant. Ignoring the 'boundary effect', the corresponding alternate route query packet failure probability at the source is given by

$$P^{(1)}_{F(1)} = \sum_{i=1}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} p_b^i \equiv 1 - P^{(1)}_{S(1)} \qquad (16)$$

where p_b is given by Equation (15), and $P^{(1)}_{S(1)}$ is the successful query packet forwarding probability.

For the remaining nodes along the source-to-destination route, the query packet is successfully forwarded if the intermediate node has at least two local neighbors (including the upstream node, from where the query packet is received), and at least one of the downstream local neighbors is available for the call. The query failure probability at the kth intermediate node, which is independent of k, for $k = 2, 3, \ldots, K$, in a K-hop route is given by

$$\begin{aligned} P^{(1)}_{F(k)} &= (N-1)\left(\frac{a}{A}\right)\left(1 - \frac{a}{A}\right)^{N-2} \\ &\quad + \sum_{i=2}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \\ &\quad \times (1 - p_b) p_b^{i-1} \equiv 1 - P^{(1)}_{S(k)} \end{aligned} \qquad (17)$$

where $P^{(1)}_{S(k)}$ is the probability of successful query packet forwarding at an intermediate node.

Therefore, with maximum *one* query forwarding neighbor, a K-hop alternate route search is successful with probability

$$P^{(1)}_S(K) = P^{(1)}_{S(1)} \left[P^{(1)}_{S(k)}\right]^{K-1} \qquad (18)$$

Case 2: *More than one forwarding neighbor:*

For more than one forwarding neighbor, the probability of successful query packet forwarding from the source to the next node is given by Equation (16), that is, $P^{(M)}_S(1) = P^{(1)}_{S(1)}$. But the probability of query success at an intermediate stage increases. An example of route query forwarding with maximum *two* forwarding nodes is shown in Figure 9. With maximum M forwarding nodes, query success probability for up to 2-hop is given by

$$\begin{aligned} P^{(M)}_S(2) &= \sum_{j=1}^{N-1} \text{Pr.}[j \text{ first-hop forwarding nodes available}] \\ &\quad \times \text{Pr.}[\text{at least } one \text{ second-hop forwarding node available}] \end{aligned}$$

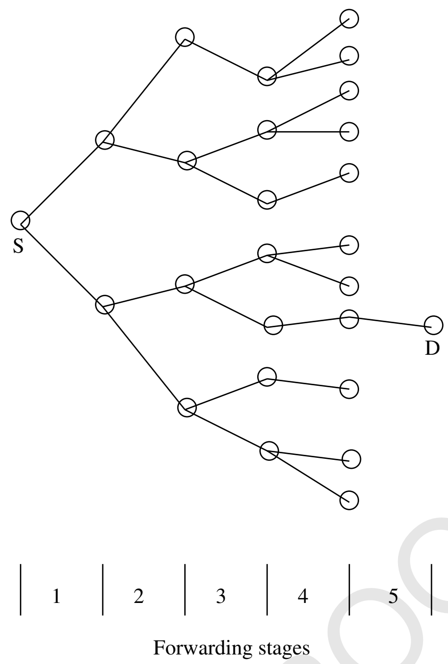


Fig. 9. An example of route request branching process with maximum *two* forwarding nodes.

$$\begin{aligned} &= \sum_{j=1}^{M} \sum_{i=j}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \\ &\quad \times C_j^i (1-p_b)^j p_b^{i-j} \times \left[1 - \left(P^{(1)}_{F(k)}\right)^j\right] \\ &\quad + \sum_{j=M+1}^{N-1} \sum_{i=j}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \\ &\quad \times C_j^i (1-p_b)^j p_b^{i-j} \times \left[1 - \left(P^{(1)}_{F(k)}\right)^M\right] \end{aligned} \qquad (19)$$

where $P^{(1)}_{F(k)}$ is obtained from Equation (17).

For obtaining the probability of query success up to 3-hop, $P^{(M)}_S(3)$, we note that from the end of stage k to the end of stage $k+2$, for all $k > 0$, the query success probability is

$$\begin{aligned} P^{(M)}_{S(k)}(2) &= \sum_{j=1}^{M} \sum_{i=j+1}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \\ &\quad \times C_j^{i-1} (1-p_b)^j p_b^{i-1-j} \times \left[1 - \left(P^{(1)}_{F(k)}\right)^j\right] \\ &\quad + \sum_{j=M+1}^{N-2} \sum_{i=j+1}^{N-1} C_i^{N-1} \left(\frac{a}{A}\right)^i \left(1 - \frac{a}{A}\right)^{N-1-i} \end{aligned}$$

$$\times C_j^{i-1}(1-p_b)^j p_b^{i-1-j} \times \left[1-\left(P_{F(k)}^{(1)}\right)^M\right] \quad (20)$$

and correspondingly, $P_{F(k)}^{(M)}(2) = 1 - P_{S(k)}^{(M)}(2)$.

Taking into account the first-hop query success probability, we have

$$P_S^{(M)}(3) = \sum_{j=1}^{M}\sum_{i=j}^{N-1} C_i^{N-1}\left(\frac{a}{A}\right)^i\left(1-\frac{a}{A}\right)^{N-1-i} \times C_j^i(1-p_b)^j p_b^{i-j} \times \left[1-\left(P_{F(k)}^{(M)}(2)\right)^j\right] + \sum_{j=M+1}^{N-1}\sum_{i=j}^{N-1} C_i^{N-1}\left(\frac{a}{A}\right)^i\left(1-\frac{a}{A}\right)^{N-1-i} \times C_j^i(1-p_b)^j p_b^{i-j} \times \left[1-\left(P_{F(k)}^{(M)}(2)\right)^M\right] \quad (21)$$

Query success probability up to 4-hop is obtained using Equations (20 and 21), with $P_{F(k)}^{(1)}$, $P_{S(k)}^{(M)}(2)$, and $P_S^{(M)}(3)$ replaced by, respectively, $P_{S(k)}^{(M)}(2)$, $P_{S(k)}^{(M)}(3)$, and $P_S^{(M)}(4)$. Higher-hop routes can be dealt with similarly.

Table IV shows the effect of the maximum number of query forwarding nodes on the alternate route query failure probability. We observe that query performance is quite stable beyond maximum *two* forwarding nodes. Therefore, without affecting call dropping performance, rerouting overhead can be minimized with maximum query forwarding nodes set to *two*.

From Table V we observe that although for maximum *one* forwarding node, query failure probability increases with distance, for maximum *two* forwarding nodes, query failure probability is almost stable beyond 2-hop from source. Intuitively, with only one forwarding node, a query process can fail at any stage, causing higher failure probability for longer distance. But, with maximum two forwarding nodes, as the depth of route search increases, there are many possible alternate routes to the destination (see Figure 9). Beyond the first two hops, query failure (or, alternatively, success) probability practically does not change. Therefore, for $M > 1$, a K-hop query success probability can be approximated as

$$P_S^{(M)}(K) \approx P_S^{(M)}(2). \quad (22)$$

From Figure 9, it is worth noting that in contrast to the geographic forwarding in GPSR [2], having more than one query forwarding node, the location information of the destination node need not be very precise, as the query process covers a zone around the anticipated location of the destination. Also, even if the location information of the local neighbors of a querying node are imprecise, the relative positions are expected to be more accurate, which are sufficient in appropriate selection of query forwarding nodes.

Table IV. Effect of maximum number of query forwarding nodes (M) on query failure probability, 2-hops away from source.

M	Query failure probability	
	$p_b = 0.2$	$p_b = 0.6$
1	3.1403×10^{-4}	0.0163
2	6.5055×10^{-5}	0.0102
3	6.4991×10^{-5}	0.0101
4	6.4991×10^{-5}	0.0101

Table V. Query failure probability with hop distance.

p_b	M	Query failure probability			
		1-hop	2-hop	3-hop	4-hop
.2	1	6×10^{-5}	3.1×10^{-4}	5.7×10^{-4}	8.2×10^{-4}
	2	6×10^{-5}	6.5×10^{-5}	6.5×10^{-5}	6.5×10^{-5}
.6	1	0.0098	0.0163	0.0228	0.0292
	2	0.0098	0.0102	0.0109	0.0109

5.3. Rerouting Control Overhead

We now determine an approximate rerouting control overhead (in terms of the number of nodes involved) associated with selective forwarding and broadcast, respectively.

Selective forwarding: For a K-hop route search, the number of nodes involved (excluding the source node) in selective forwarding with M forwarding neighbors is upper-bounded as

$$n_{\text{sel}}^{(K)}(M) \le 1 + M + M^2 + \cdots + M^{K-1} \quad (23)$$

where the equality holds if at all forwarding stages at least M potential forwarding nodes are available. Thus, for a K-hop route search with $M = 2$, the maximum number of nodes involved is $2^K - 1$.

Broadcast: Since the nodes are uniformly randomly distributed in the mobility space, the average geographical distance of a K-hop source-to-destination route (along the shortest path) is approximately $K\frac{R}{2}$.

Therefore, ignoring the ‘boundary effect’, the overhead involved in broadcast-based route discovery is approximately the number of nodes around the source within the circle of radius $K\frac{R}{2}$, which is given by

$$n_{\text{bcast}}^{(K)} \approx \left(\frac{\pi K^2 R^2}{4}\right)\left(\frac{N}{A}\right) = \frac{Na}{4A}K^2 \tag{24}$$

where $a = \pi R^2$ and N is the total number of nodes in the mobility space of area A (see discussions on Equation 14).

Estimates of the number of nodes involved in selective forwarding and broadcast-based rerouting approaches, along with the route length, expected route lifetime (derived in Section 4), and session duration, enable us to compare the rerouting overheads in these two cases. Numerical results will be presented in the next section.

6. Simulation Experiments

Performance of the proposed TDR protocol is studied via C-based discrete event simulation. We are primarily interested in studying the *effect of mobility* on selective forwarding and prediction-based distributed routing, and comparing them with broadcast-based and reactive routing schemes. For simplicity, channel-fading effects are not included in our current simulation, which will affect all the routing schemes discussed here, but not the general performance trends.

As discussed in the mobility model (Section 4), the mobile hosts (also called *users* or *nodes*) are assumed to be• ‘well behaved’ such that their movement patterns are not completely random. In our simulations, a node’s average velocity in an epoch** is constant along a specific direction (for both velocity profiles). At the end of an epoch, the velocity and movement direction of the node randomly changes only within certain limits. To trigger an alternate route search, in addition to the current receive power (based on relative distance), we take into account the rate of change of receive power. This is to ensure some priority to the active nodes with degrading link condition [27]. Only when the current receive power is below a predefined lower threshold and its rate of change is negative, the alternate route discovery process is initiated.

The following assumptions on the network condition are made in the simulation: (i) Poisson arrival process; (ii) exponentially distributed session duration; (iii) equiprobable source–destination pairs; (iv) a node can handle more than one session simultaneously; (v) only real-time applications are considered. Since only real-time sessions are considered, an in-session data flow is always along a preset route. Because of this, in-session MAC conflict is assumed to be• nonexistent. It is also ensured that no nodal or network partition occurs during run time. Since the fading-channel effect is not included, the receive power is considered in terms of equivalent internodal distance.

The values considered for the simulation parameters are as follows: area of mobility space, $A = 1500 \times 1000$ m^2; default number of nodes, $N = 60$; range of a mobile node, $R = 300$ m; end threshold distance $Th_2 = 270$ m; average nodal velocity 1 m s^{-1} to 10 m s^{-1}; maximum velocity change per epoch 10% of average; maximum direction change per epoch (uniformly distributed) 90°; maximum data-handling capacity of a node 10 kbps; maximum data rate per session (uniformly distributed) 2 kbps; average session interarrival time per node, $1/\lambda = 6$ min; default average session duration, $\bar{x} = 3$ min; average epoch length 6 sec; default maximum number of query forwarding nodes from a node, $M = 2$. Sufficient number of sessions are attempted to attain the simulation results within 95% confidence interval.

On the basis of the above assumptions and parameter values, we study the network performance with the proposed TDR protocol and compare it with three existing QoS routing protocols, for example, FORP, DQoSR, and E-AODV.

In evaluating and comparing the TDR protocol performance, it is assumed that with insufficient resource, an attempted session could be either lost (blocked call lost (BCL) model) or delayed (blocked call delayed (BCD) model). In the BCL model, the session acceptance performance is measured by *grade of service* (GoS), which is the ratio of the sum of blocked and dropped sessions to the number of attempts. In the BCD model, the session acceptance performance is measured by *queueing delay*, which is the average waiting time of an attempted session in the input buffer before it is accepted.

Because the network topology and mobility pattern vary widely for different SEED values, for each

** An epoch is specified by the session interarrival time in the network.

protocol we simulate six different scenarios for each average velocity.

In the following text, first, we verify the analysis on route lifetime and rerouting control overhead. Then, we study separately the performance of the proposed TDR protocol, for example, GoS and control overhead for different selective forwarding cases. Finally, the comparative performance results of TDR with respect to the other existing proposals (e.g. FORP, DQoSR, and E-AODV) are carried out.

6.1. Verification of the Analysis

We study the variation of route lifetime in the simulated mobility model for uniformly distributed (Figure 10) and Rayleigh-distributed (Figure 11)

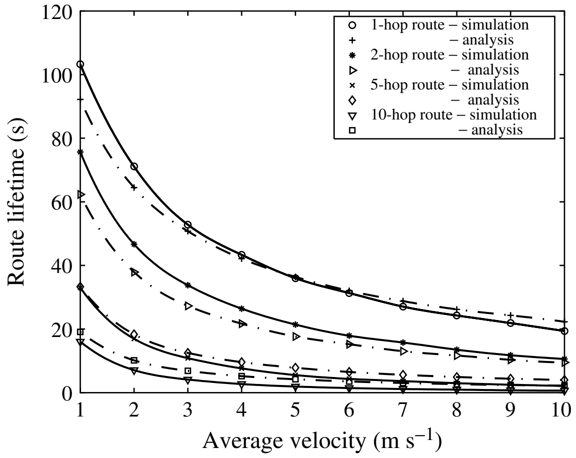


Fig. 10. Route lifetime for uniformly distributed velocity profile.

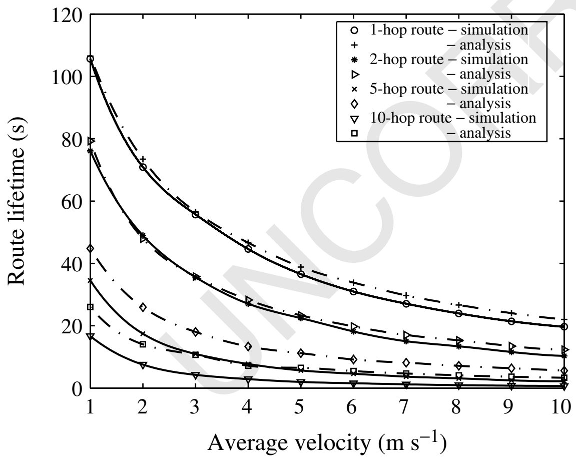


Fig. 11. Route lifetime for Rayleigh-distributed velocity profile.

velocity profiles and compare them with the numerical results from analysis. In both cases, trends of route lifetime from simulation are quite similar to that from analysis. Particularly, for lower mobility and shorter route length, the match between simulation and analysis is quite good. For higher mobility and longer route, the active mobile nodes hit the boundary of rectangular mobility space more frequently, which disrupt the normal mobility pattern in simulation, leading to more route failure and hence shorter lifetime. In our analytic mobility models, velocity and direction changes at the session rerouting instants have no correlation with the respective previous states, whereas in simulation new velocity and direction at every epoch are correlated with their respective previous values. This fact may explain the differences of simulation results from analysis at low mobility.

Figure 12 shows the ECOH plots for full broadcast-based and selective forwarding-based route discovery for uniformly distributed velocity profile. If $n_{\text{rr}}^{(K)}$ is the number of nodes visited in a K-hop rerouting process, $\overline{T}_R$ is the average route lifetime (given by Equation (9) or (13)), and $\overline{x}$ is the average call duration, then ECOH is defined as $ECOH = (\overline{x}/\overline{T}_R)n_{\text{rr}}^{(K)}$. For selective forwarding, $n_{\text{rr}}^{(K)} = n_{\text{sel}}^{(K)}(M)$ (Equation 23); for full broadcast, $n_{\text{rr}}^{(K)} = n_{\text{bcast}}^{(K)}$ (Equation 24). n_g is obtained from Equation (14). The analytic results are obtained for $M = 2$. The simulated results match closely the analytic prediction. The deviation at higher velocities could be because of more frequent route disruption in simulation due to the bordering nodes' mobility. The comparative ECOH plots for Rayleigh-distributed velocity profile follow similar trends as in Figure 12.

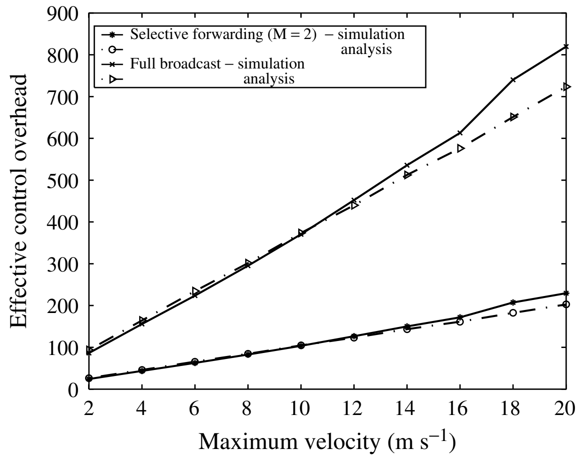


Fig. 12. Effective control overhead for uniformly distributed velocity profile for a 3-hop route.

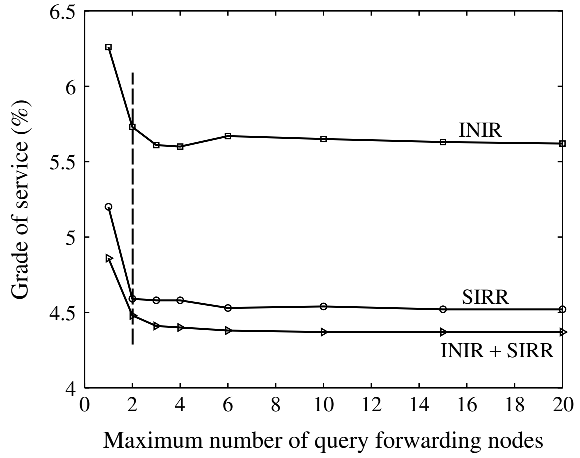


Fig. 13. Grade of service (sum of call blocking and dropping rates) variation with maximum number of query forwarding nodes. Maximum velocity 2 m s^{-1}.

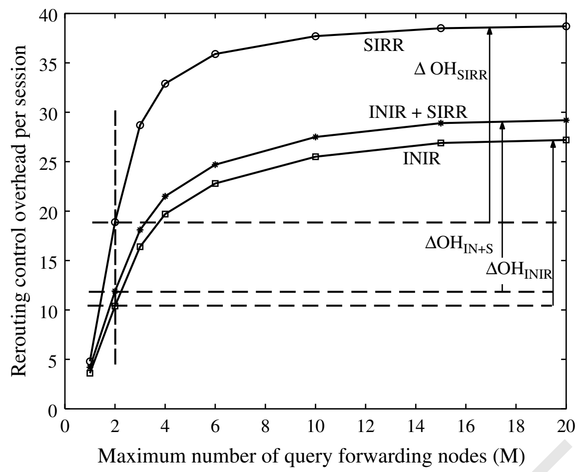


Fig. 14. Control overhead variation with maximum number of query forwarding nodes. Maximum velocity 2 m s^{-1}. ΔH's are the respective additional control overhead required for full broadcast-based alternate route search, without substantial gain on grade of service.

In the subsequent discussions on comparative performance results, uniform velocity profile is considered.

6.2. TDR Protocol Evaluation

Although the TDR protocol adopts distributed rerouting, for comparing its overhead with respect to the end node–controlled rerouting, namely, SIRR (e.g. in [10,17]), we study INIR as well as SIRR. Here, we consider that if at any time the current route fails and an alternate route could not be found *a priori* (via INIR, SIRR, or INIR followed by SIRR), the session is dropped.

Figures 13 and 14 show, respectively, GoS (call blocking + dropping rate) and rerouting control overhead variation versus the number of reroute request forwarding nodes, for uniformly distributed velocity profile, with maximum velocity 2 m s^{-1}. Rerouting control overhead is a measure of the average number of rerouting requests forwarded per session.

First, we observe that in all cases (SIRR, INIR, and INIR + SIRR), GoS remains steady beyond maximum *two* reroute request forwarding nodes (i.e. $M = 2$), while the control overhead continues to increase for up to approximately $M = 11$ (which is nearly the average number of neighbors). The GoS does not decrease further beyond $M = 2$, because for $M > 2$, service degradation is mostly due to mobility-dependent failure. Beyond the saturation level of GoS, the additional overheads due to full broadcast with SIRR, INIR, and INIR + SIRR, ΔOH_{SIRR}, ΔOH_{INIR}, and $\Delta OH_{\text{IN,S}}$, respectively, are also indicated in the figure, which indicate the utility of selective forwarding without losing the GoS performance. These results also verify the analytic results (Tables IV and V). Second, it is observed that INIR alone performs a little poorer over SIRR in terms of GoS. As a reason, we note that although the average searching distance in INIR is shorter (which will require lesser searching time) compared to the SIRR, rerouting from intermediate nodes causes (end-to-end) longer routes (see Figure 15), associated with higher failure. Particularly, if the query-initiating node is very close to the destination, INIR may fail to secure an alternate route. In TDR, if at any point INIR fails, the rerouting control is transferred to the source, that is, then INIR is followed by SIRR. The GoS and control overhead plots show that INIR + SIRR has even better GoS performance and yet lesser control overhead compared to SIRR. Thus, INIR + SIRR takes the advantage of distributed rerouting control

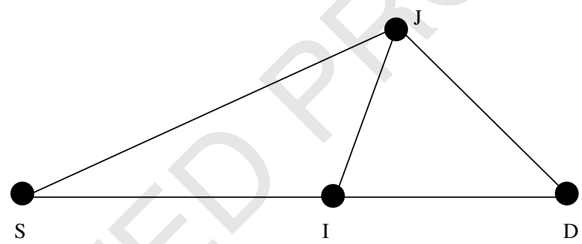


Fig. 15. An example of triangular route selection. The original route is S-I-D. The new route for SIRR is S-J-D, whereas that for INIR is S-I-J-D. From triangle law, $length\ (\text{SI}) + length\ (\text{IJ}) > length\ (\text{SJ})$.

(INIR) and a possible second opportunity of alternate route search (SIRR following INIR).

Figures 16 and 17 show the GoS and associated control overhead variation with mobility. It is further observed here that INIR + SIRR performs as well as SIRR, and yet requires lesser control overhead. Unlike in Figure 12, the control plots in Figure 17 do not resemble linear.● This is because, control overhead for longer routes reach saturation values (due to 'boundary effect'), thereby introducing nonlinearity in the cumulative average for different route lengths. The additional threshold margin (and associated time) required in INIR + SIRR and its effect on network performance is not within the scope of our current simulation.

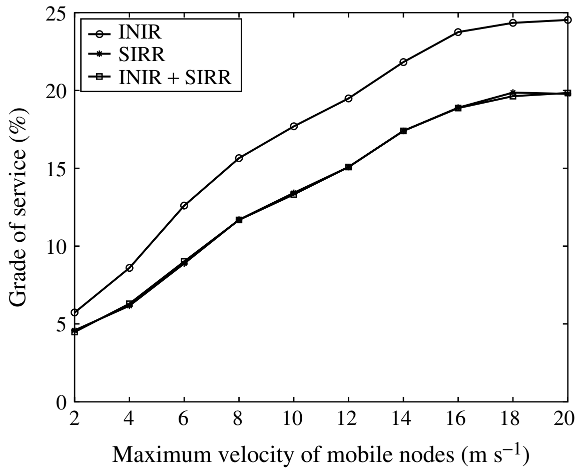


Fig. 16. Grade of service versus nodal mobility (maximum *two* query forwarding nodes).

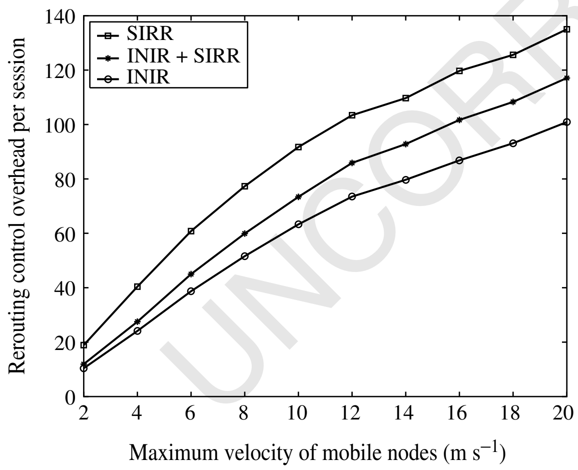


Fig. 17. Control overhead versus nodal mobility (maximum *two* query forwarding nodes).

6.3. Comparison Results

In comparing the TDR protocol performance with those of FORP, DQoSR, and E-AODV, it is assumed that once a session is successfully initiated, it is not dropped prematurely even if there is intermittent route failure. The packets during the route failure intervals are dropped. The protocol performance in such cases is measured in terms of *QoS ratio*, which is defined as the fractional successful packet transmissions per session, or alternatively as packet dropping probability. Note that since in all protocols the neighborhood/network information is maintained by periodic beaconing, this common overhead is not taken into account for comparison of control overhead; rather only the rerouting overheads are considered.

In FORP [10], only one active route is maintained. On the basis of the predicted route failure time, the destination initiates *broadcast-based* alternate route discovery up to the source. From the rerouting control point of view, this scheme is similar to TDR with SIRR (with $M \gg 2$).

In simulating DQoSR protocol [7] we consider up to two disjoint routes (the primary and one secondary). A session is accepted even if only one (primary) route could be secured. At any stage, if a session has only the primary route, the source tries for a secondary route at every status update epoch. In case of primary route failure, if there is a secondary route available, it immediately takes over the session and is treated as the current primary route. There is no QoS degradation in this case. On the other hand, during primary route failure, if no secondary route exists, the packets are dropped as long as the route failure persists.

In E-AODV protocol [9], only the active routes are maintained (soft-state concept). No attempt is made to maintain the source-to-destination logical connection. If the route fails, *broadcast-based* route discovery process is reinitiated from the source. The packets during the route failure intervals are dropped.

We provide the comparative performance results of these four protocols (TDR, FORP, DQoSR, and E-AODV) for the BCD model. The results for the loss model are not shown as they follow similar trends.

Figure 18 shows the QoS performances of different protocols, where it is observed that the E-AODV performs poorly at higher velocity as it has neither route prediction capability nor does it maintain alternate routes. TDR and FORP perform nearly the same as both protocols operate under the same prediction capability. DQoSR performs a little poorer than TDR and FORP, since it has to allocate more resources to

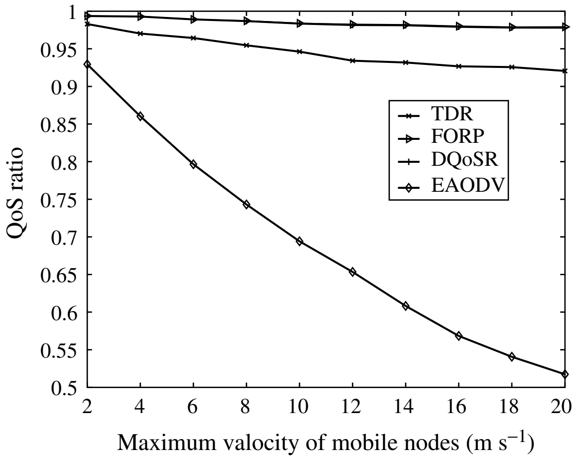


Fig. 18. Variation of QoS ratio with mobility.

support the ongoing sessions and also because of its reactive nature.

Figure 19 shows the average control overhead per session (average number of rerouting control packets generated per successful session) associated with the protocols. Since DQoSR maintains secondary resources for the ongoing sessions, the sessions experience the minimum overhead, but the total overhead experienced by the network is much higher. Note that DQoSR has an additional nodal database overhead for maintaining network-wide delay and bandwidth information, which is not captured in our simulation. E-AODV has higher control overhead than that seen by a session in DQoSR because in this case every time the route fails, the session is interrupted and it (E-AODV) has to immediately start an alternate route discovery process. Distributed rerouting control and selective forwarding-based route discovery causes lesser rerouting overhead in TDR than that in FORP, which adopts localized control and broadcast-based route discovery. Although both FORP and E-AODV follow broadcast-based route search, FORP being proactive protocol requires more frequent invocation of rerouting routine, leading to higher overhead compared to E-AODV.

Variation of average rerouting control overhead per session with network size (for nearly the same nodal density, by varying the area of mobility space with the number of nodes) is shown in Figure 20. Call arrival rate at each node is kept constant for different network size. The obvious general trend is that the average route length increases with increase in network size, causing increase in route maintenance overhead. It also shows that TDR has low rate of overhead increment. Although FORP maintains only the active route, its broadcast-based route discovery causes higher overhead increment rate. The control overhead in DQoSR is lower than the case of FORP, as the route discovery is controlled by the number of tickets. Having poor QoS support in E-AODV, its overall control overhead is also low and the increment is slower.

QoS ratio versus average route length plot is shown in Figure 21, where for the same nodal density, the average source-to-destination distance is explicitly varied by changing the length-to-breadth ratio of the rectangular mobility space. Here also it is observed that proactively rerouting (in TDR and FORP) enables maintaining the logical route better. Again, E-AODV has much faster QoS ratio degradation, as it has neither link failure prediction mechanism nor does it maintain any alternate route.

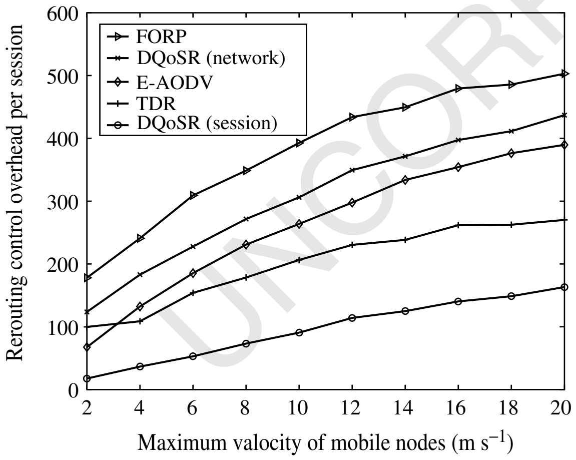


Fig. 19. Rerouting control overhead at different mobility.

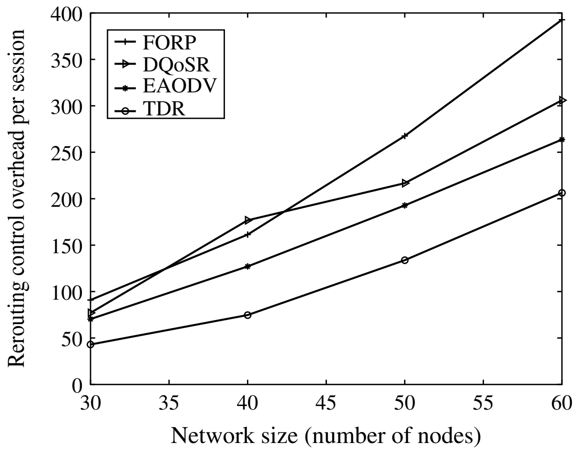


Fig. 20. Rerouting control overhead versus network size. Maximum velocity 10 m s^{-1}.

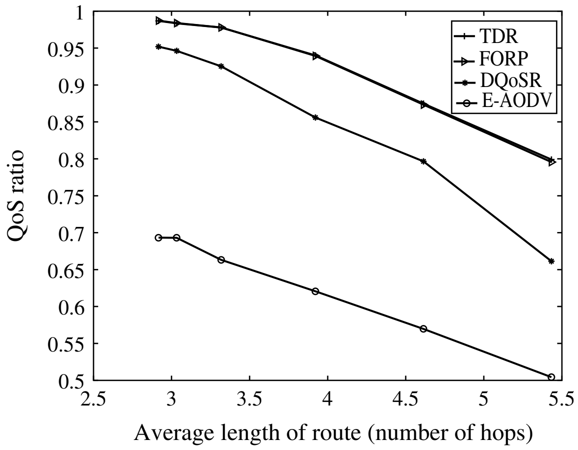


Fig. 21. QoS ratio versus source-to-destination distance (average hop length). Number nodes: 60; Maximum velocity 10 m s^{-1}.

The preemptive routing approach in [17] was not explicitly considered for comparison as it is similar to FORP. Particularly, both these protocols follow end node–controlled rerouting (FORP is destination-controlled, whereas preemptive routing is source-controlled) and both of them do not use location information in the alternate route discovery process.

7. Concluding Remarks

In this paper, we have presented a routing scheme called trigger-based distributed routing (TDR) for supporting RT-QoS traffic in mobile ad hoc networks. The proposed TDR scheme uses failure prediction-based alternate route discovery and avoids maintenance of additional routes. This reduces control traffic as well as the size of nodal database. In addition, TDR makes use of selective forwarding of routing requests based on relative location information, and as a result its route discovery overhead is further reduced. As an added cost, this protocol requires some extra nodal computation for selecting appropriate nodes to forward route requests.

Analytic mobility models have been developed to estimate the route lifetime and associated rerouting control overhead for RT-QoS support. The effect of selective forwarding on rerouting success is quantified via analysis. Simulations have been conducted and the results have been verified to follow closely with the analytic results.

The TDR protocol performance has been studied and compared with the existing QoS protocols for ad hoc networks, such as FORP, DQoSR, and E-AODV via simulations. Significant superiority in the QoS performance of 'prediction-based' TDR over these 'prediction-less' QoS routing protocols (E-AODV, DQoSR) has been noted. Both TDR and FORP are 'prediction-based' protocols, and both have a comparable QoS performance in terms of queueing delay and QoS ratio. But, having distributed control and selective packet forwarding, TDR requires limited control overhead and has better scalability.

In the simulation, to ensure full QoS support, whenever logical flow paths were available, resource reservations were done on maximum bandwidth demand for a session. This model can be extended to study the QoS performance based on the minimum bandwidth demand (for flexible QoS support) and with heterogeneous traffic. In such cases, however, even if a flow path exists, there can be QoS degradation in terms of QoS ratio and end-to-end delay variation due to burstiness of packet arrivals. The fading-channel effect has not been considered as we are primarily interested in studying the benefit of proactive and selective forwarding-based rerouting over reactive and broadcast-based rerouting strategies. Since channel fading will affect the performance of all the protocols, we expect that the trends of performance results will remain valid.

Appendix I

Probability of a node having *i* neighbors

Here we provide a more accurate estimate of the number of neighbors of a node for uniformly distributed nodes within a rectangular space. The coverage region of a mobile node is assumed to be• circular with radius R. For any other regular mobility space (e.g. circular), a similar approach has to be devised for the estimate. Q14

The probability of a node having i neighbors is in general given by

$$P(i) = \sum_{(x,y)\in(X,Y)} P(x,y) C_i^{N-1} \left(\frac{a(x,y)}{A}\right)^i \times \left(1 - \frac{a(x,y)}{A}\right)^{N-1-i} \qquad \text{(A.1)}$$

where $P(x, y)$ is the probability of finding the node under consideration at the point (x, y), and $a(x, y)$ is the area covered by the node within the mobility space.

Referring to Figure A.1, the probability of having i neighbors around a node is dependent on the node's

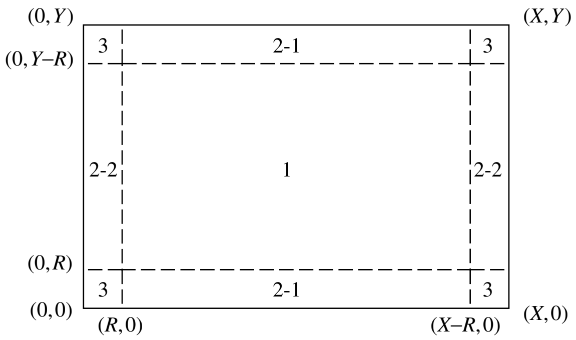


Fig. A.1. A rectangular mobility space showing different zones.

location. Depending on a node's coverage within the rectangular region, we divide the rectangular region into three zones—zone 1, zone 2, and zone 3. For example, if the node is in zone 1, its entire coverage lies within the rectangle. Whereas if the node is in zone 3, its coverage within the rectangle varies for every different position.

Zone 1:

$$R < x < X - R, R < y < Y - R$$

Here $a(x, y) = \pi R^2$, a constant, is denoted as a_1. Denoting $A = XY$, the probability of having i nodes in this case is obtained as

$$P_1 = \frac{(X - 2R)(Y - 2R)}{A} C_i^{N-1} \left(\frac{a_1}{A}\right)^i \times \left(1 - \frac{a_1}{A}\right)^{N-1-i} \quad \text{(A.2)}$$

Zone 2:

Case 1 (Zone 2-1): $R < x < X - R, y < R$

Referring to Figure A.2, the area covered within the rectangle (a_{21}) is given by *ABCDEA*. The probability of having i nodes in this case is obtained as

$$P_{21} = \frac{1}{A} \sum_{\substack{R<x<X-R \\ y<R}} C_i^{N-1} \left(\frac{a_{21}}{A}\right)^i \times \left(1 - \frac{a_{21}}{A}\right)^{N-1-i} \quad \text{(A.3)}$$

where $a_{21} = \left\{\pi - \tan^{-1}\left(\frac{\sqrt{R^2 - y^2}}{y}\right)\right\} R^2 + y\sqrt{R^2 - y^2}$. Note that the area a_{21} and P_{21} being

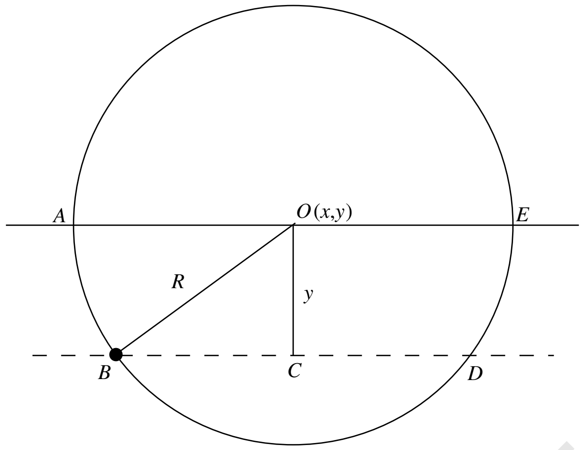


Fig. A.2. Location of the node showing partial coverage within the mobility space (Area *ABCDEA*).

(x, y)-dependent, P_{21} has to be computed by numerical simulation.

Case 2 (Zone 2-2): $x < R, R < y < Y - R$

The probability of having i nodes in this case is

$$P_{22} = \left(\frac{1}{A}\right) \sum_{\substack{x<R \\ R<y<Y-R}} C_i^{N-1} \left(\frac{a_{22}}{A}\right)^i \times \left(1 - \frac{a_{22}}{A}\right)^{N-1-i} \quad \text{(A.4)}$$

where a_{22} is obtained similarly as in the case of *Zone 2-1*. Here also P_{22} has to be computed by numerical simulation.

From Equations (A.3 and A.4), total probability for the node in *zone 2* having i neighbors is

$$P_2 = 2\,(P_{21} + P_{22}) \quad \text{(A.5)}$$

Zone 3:

We consider the zone corresponding to the corner point coordinate (0, 0).

Case 1: $(x - R)^2 + (y - R)^2 \leq R^2, x, y \geq 0$

Refer to Figure A.3. Area (a_{31}) covered within the rectangle by the node is the area $AA_1A_2PQA_4A_3A =$ Area(AA_1CA_3) + Area(A_1CA_2) + Area(A_3CA_4) +

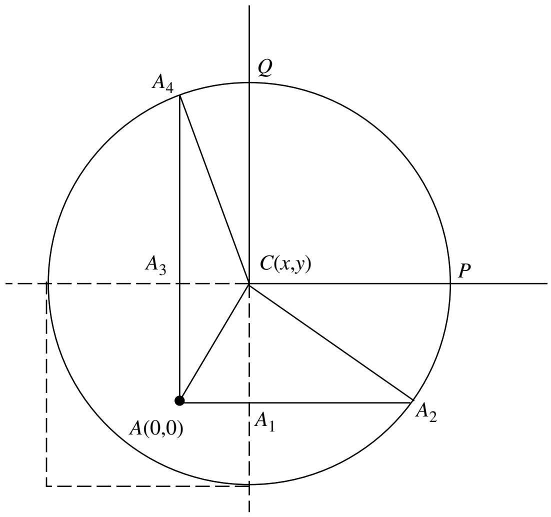


Fig. A.3. Location of the node showing partial coverage within the mobility space (Area $AA_1A_2PQA_4A_3A$).

$\mathrm{Area}(A_2CP) + \mathrm{Area}(A_4CQ) + \mathrm{Area}(PCQ)$, which is

$$a_{31} = \frac{\pi R^2}{4} + xy + \frac{y}{2}\sqrt{R^2 - y^2} + \frac{x}{2}\sqrt{R^2 - x^2} + \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{R^2 - x^2}}{x}\right)\right]\frac{R^2}{2} + \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{R^2 - y^2}}{y}\right)\right]\frac{R^2}{2} \tag{A.6}$$

The probability of having i nodes in this case is

$$P_{31} = \left(\frac{1}{A}\right) \sum_{\substack{(x-R)^2+(y-R)^2 \le R^2 \\ x,y \ge 0}} C_i^{N-1} \left(\frac{a_{31}}{A}\right)^i \times \left(1 - \frac{a_{31}}{A}\right)^{N-1-i} \tag{A.7}$$

Case 2: $R^2 < (x - R)^2 + (y - R)^2 \le 2R^2$, $x, y \ge 0$.

Refer to Figure A.4. The area (a_{32}) covered within the rectangle by the node is the area $B'B_1B_2B_3PQB_6B_5B_4B' = \mathrm{Area}(B_1CB_3B_2B_1) + \mathrm{Area}(B_3CP) + \mathrm{Area}(PCQ) + \mathrm{Area}(B_6CQ) + \mathrm{Area}(B_4B_5B_6CB_4) + \mathrm{Area}(B_1B'B_4CB_1)$, which is

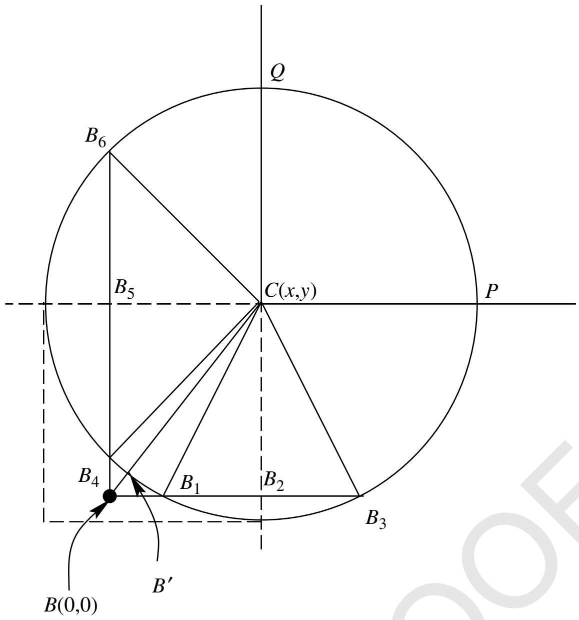


Fig. A.4. Location of the node showing partial coverage within the mobility space (Area $B'B_1B_2B_3PQB_6B_5B_4B'$).

$$a_{32} = \frac{\pi R^2}{4} + x\sqrt{R^2 - x^2} + y\sqrt{R^2 - y^2} + \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{R^2 - x^2}}{x}\right)\right]\frac{R^2}{2} + \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{R^2 - y^2}}{y}\right)\right]\frac{R^2}{2} + \left[\frac{\pi}{2} - \tan^{-1}\left(\frac{\sqrt{R^2 - x^2}}{x}\right) - \tan^{-1}\left(\frac{\sqrt{R^2 - y^2}}{y}\right)\right]\frac{R^2}{2} \tag{A.8}$$

The probability of having i nodes in this case is

$$P_{32} = \left(\frac{1}{A}\right) \sum_{\substack{R^2 < (x-R)^2+(y-R)^2 \le 2R^2 \\ x,y \ge 0}} C_i^{N-1} \left(\frac{a_{32}}{A}\right)^i \left(1 - \frac{a_{32}}{A}\right)^{N-1-i} \tag{A.9}$$

As in the case of *zone 2*, P_{31} and P_{32} have to be computed by numerical simulation.

From Equations (A.7 and A.9), total probability for the node in *zone 3* having i neighbors is

$$P_3 = 4\,(P_{31} + P_{32}) \tag{A.10}$$

Finally, from Equations (A.2, A.5, and A.10), the probability of a node having i neighbors is given by

$$P(i) = P_1 + P_2 + P_3 \tag{A.11}$$

Acknowledgments

This work was partially supported by Nortel Networks, Richardson, TX. This paper was presented in part as a poster at the MobiHoc 2002 conference, Lausanne, Switzerland, June 2002.

References

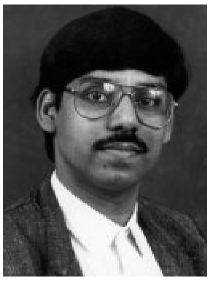
1. Johnson DB, Maltz DA. Dynamic source routing in ad hoc wireless networks. Chap. 5. In *Mobile Computing*. Imielinski T, Korth H (eds). Kluwer Academic Publishers: 1996, pp. 153–181.
2. Karp B, Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of ACM MobiCom*, 2000.
3. Park V, Corson MS. Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification. IETF Internet Draft, draft-ietf-ma-net-tora-spec-00.txt, December 1997.
4. Pearlman MR, Haas ZJ. Determining the optimal configuration for the zone routing protocol. *IEEE Journal on Selected Areas in Communications* 1999; **17**(8): 1395–1414.
5. Perkins CE, Royer EM. Ad hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.
6. Toh CK. Associativity-based routing for ad hoc mobile networks. *Wireless Personal Communications Journal* 1997; **4**(2): 103–139.
7. Chen S, Nahrstedt K. Distributed quality-of-service routing in ad hoc networks. *IEEE Journal of Selected Areas in Communications* 1999; **17**(8): 1488–1505.
8. Perkins CE, Bhagwat P. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of ACM SIGCOMM*, 1994, pp. 234–244.
9. Perkins CE, Royer EM, Das SR. Quality of Service for Ad Hoc On-Demand Distance Vector Routing. IETF Internet Draft, draft-ietf-manet-aodvqos-00.txt, July 2000.
10. Su W, Gerla M. IPv6 flow handoff in ad hoc wireless networks using mobility prediction. In *Proceedings of IEEE GLOBECOM*, 1999.
11. Ko YB, Vaidya NH. Location-aided routing (LAR) in mobile ad hoc networks. In *Proceedings of ACM MobiCom*, 1998.
12. Murthy S, Garcia-Luna-Aceves JJ. An efficient routing protocol for wireless networks. *Mobile Networks and Applications* 1996; **1**(2): 183–197.
13. Chen TW, Gerla M. Global state routing: a new routing scheme for ad-hoc wireless networks. In *Proceedings of IEEE ICC*, 1998, pp. 171–175.
14. Basagni S, Chlamtac I, Syrotiuk VR, Woodward BA. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of ACM MobiCom*, 1998.
15. Lee SB, Ahn GS, Zhang X, Campbell AT. INSIGNIA: an IP-based quality of service framework for mobile ad hoc networks. *Journal of Parallel and Distributed Computing* 2000; **60**: 374–406.
16. Maltz DA. Resource Management in Multi-hop Ad Hoc Networks. Technical Report, CMU-CS-00-150. School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, November 2000.
17. Goff T, Abu-Ghazaleh N, Pathak D, Kahvecioglu R. Preemptive routing in ad hoc networks. In *Proceedings of ACM MobiCom*, 2001.
18. Nasipuri A, Das SR. On-demand multipath routing for mobile ad hoc networks. In *Proceedings of International Conference on Computer Communications and Networks (IC3N)*, 1999, pp. 64–70.
19. ATM Forum. Private Network Network Interface (PNNI). af-pnni-0055.000, April 2002, Version 1.1.
20. Ward A, Jones A, Hopper A. A new location technique for the active office. *IEEE Personal Communications* 1997; **4**(5): 42–47.
21. http://www.ietf.org/rfc/rfc2205.txt, September 1997.
22. Rappaport T. *Wireless Communications: Principles and Practice*. Prentice Hall: New York, 1996.
23. Hong D, Rappaport SS. Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures. *IEEE Transactions on Vehicular Technology* 1986; **35**(3): 77–92.
24. McDonald AB, Znati T. A mobility based framework for adaptive clustering in wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications* 1999; **17**(8): 1466–1486.
25. Yeung K, Nanda S. Channel management in microcell/macrocell cellular radio systems. *IEEE Transactions on Vehicular Technology* 1996; **45**(4): 601–612.
26. Zonoozi M, Dassanayake P. User mobility modeling and characterization of mobility patterns. *IEEE Journal on Selected Areas in Communications* 1997; **15**(7): 1239–1252.
27. Ebersman HG, Tonguz OK. Handoff ordering using signal prediction priority queueing in personal communication systems. *IEEE Transactions on Vehicular Technology* 1999; **48**(1): 20–35.
28. Basagni S. Distributed clustering for ad hoc networks. In *International Symposium on Parallel Architectures, Algorithms, and Networks*, Perth, June 1999,• pp. 310–315. Q15

Authors' Biographies

Swades De received the B.Tech degree in radiophysics and electronics from the University of Calcutta in 1993 and the M.Tech degree in optoelectronics and optical communication from the Indian Institute of Technology, Delhi, in 1998. During 1993–1997 and in the first half of 1999, he worked in different telecommunication companies in India as a hardware and software development engineer. He is a Ph.D candidate in the Electrical Engineering Department at State University of New York at Buffalo. His current research interests include performance study, multipath routing in high-speed networks, QoS routing and resource optimization in mobile ad hoc networks and wireless sensor networks, load balancing in cellular wireless networks, and communications and systems issues in optical networks.

Chunming Qiao is an associate professor at the University at Buffalo (SUNY), where he directs the Lab for Advanced Network Design, Analysis, and Research (LANDER) that conducts cutting-edge research work on optical networks, wireless networks, and the Internet. Dr Qiao has published more than 120 papers in leading technical journals and conference proceedings and is recognized for his pioneering research on Optical Internet, in particular, the optical burst switching (OBS) paradigm. His work on integrated cellular and ad hoc networking systems (iCAR) is also internationally acclaimed and has been featured in Businessweek and Wireless Europe.

Dr Qiao is on the editorial board of several journals and magazines including IEEE Communications and IEEE/ACM Transactions on Networking (ToN) and has guest-edited three IEEE JSAC issues. He has chaired and cochaired many international conferences and workshops including the High-Speed Networking Workshop (formerly GBN) at Infocom'01 and Infocom'02, Opticomm'02, and the symposium on Optical Networks at ICC'03.



Sajal K. Das is a professor of computer science and engineering and also the founding director of the Center for Research in Wireless Mobility and Networking (CReWMaN) at the University of Texas at Arlington (UTA). Prior to 1999, he was a professor of computer science at the University of North Texas (UNT). He is a recipient of the UNT Student Association's Honor Professor Award in 1991 and 1997 for best teaching and scholarly research, and UTA's Outstanding Senior Faculty Research Award in 2001. He has visited numerous universities and research organizations worldwide for collaborative research and invited seminar talks. His current research interests include resource and mobility management in wireless networks, mobile and pervasive computing, wireless multimedia and QoS provisioning, sensor networks, mobile Internet architectures and protocols, distributed processing and grid computing. He has published over 200 research papers in these areas and holds 4 US patents in wireless mobile networks. He received the Best Paper Awards in ACM MobiCom'99, ICOIN'02, ACM MSWIM'00, and ACM/IEEE PADS'97. Dr Das serves on the Editorial Boards of IEEE Transactions on Mobile Computing, ACM/Kluwer Wireless Networks, Computer Networks, Journal of Parallel and Distributed Computing, Parallel Processing Letters, Journal of Parallel Algorithms and Applications. He served as General Chair of IEEE MASCOTS'02, ACM WoWMoM'00-'02; General Vice Chair of IEEE PerCom'03, ACM MobiCom'00 and HiPC'00-'01; Program Chair of IWDC'02, WoWMoM'98-'99; TPC Vice Chair of ICPADS'02; and as TPC member of numerous IEEE and ACM conferences. He is a member of IEEE TCPP and TCCC Executive Committees and of the advisory boards of several cutting-edge companies.

QUERIES TO BE ANSWERED BY AUTHOR (SEE MARGINAL MARKS)

IMPORTANT NOTE: Please mark your corrections and answers to these queries directly onto the proof at the relevant place. Do NOT mark your corrections on this query sheet.

Query No.	Query
Q1	Please clarify if the details of the corresponding address are correct.
Q2	Please clarify if the abbreviations DSR, TORA, ABR, ZRP, AODV and GPSR need to be spelt out at the first instance. If so, please provide the expansions.
Q3	Please clarify if the abbreviations DSDV, WRP, GSR, DREAM needs to be spelt out at the first instance. If so, please provide the expansions.
Q4	We have rephrased this part of the sentence. Please clarify if we have retained the intended meaning.
Q5	We have changed ‘in the next subsection’ to ‘Section 3.2’. Please clarify if this is correct.
Q6	Please clarify if this abbreviation needs to be spelt out at the first instance. If so, please provide the expansion.
Q7	We have spelt out ‘LAR’ as ‘location-aided routing’. Please clarify if this is correct.
Q8	We have rephrased this part of the sentence. Please clarify if this retains the intended meaning.
Q9	We have rephrased this part of the sentence. Please clarify if this is fine.
Q10	Please spell out ‘i.i.d.’ at the first instance.
Q11	We have rephrased this part of the sentence. Please clarify if we have retained the intended meaning.
Q12	We have rephrased this part of the sentence. Please clarify if this retains the intended meaning.
Q13	Please clarify if there are any words missing in the sentence ‘Unlike in...’.
Q14	We have rephrased this part of the sentence. Please clarify if we have retained the intended meaning.
Q15	Reference 28 has not been cited in text. Please provide the place of citation.