

On-Demand Multipath Routing for Mobile Ad Hoc Networks *

Asis Nasipuri and Samir R. Das
Division of Computer Science
The University of Texas at San Antonio
San Antonio, TX 78249-0667
Email: {nasipuri, samir@cs.utsa.edu}

Abstract- Mobile ad hoc networks are characterized by multi-hop wireless links, absence of any cellular infrastructure, and frequent host mobility. Design of efficient routing protocols in such networks is a challenging issue. A class of routing protocols called *on-demand* protocols has recently found attention because of their low routing overhead. The on-demand protocols depend on query floods to discover routes whenever a new route is needed. Such floods take up a substantial portion of network bandwidth. We focus on a particular on-demand protocol, called *Dynamic Source Routing*, and show how intelligent use of multipath techniques can reduce the frequency of query floods. We develop an analytic modeling framework to determine the relative frequency of query floods for various techniques. Results show that while multipath routing is significantly better than single path routing, the performance advantage is small beyond a few paths and for long path lengths. It also shows that providing all intermediate nodes in the primary (shortest) route with alternative paths has a significantly better performance than providing only the source with alternate paths.

1 INTRODUCTION

A mobile, *ad hoc* network is an autonomous system of mobile hosts connected by wireless links. There is no static infrastructure such as base station. If two hosts are not within radio range, all message communication between them must pass through one or more intermediate hosts that double as routers. The hosts are free to move around randomly, thus changing the network topology dynamically. Thus routing protocols must be adaptive and able to maintain routes in spite of the changing network connectivity. Such networks are very useful in military and other tactical applications such as emergency rescue or exploration missions, where cellular infrastructure is unavailable or unreliable. Commercial applications are also likely where there is a need for ubiquitous communication services without the presence or use of a fixed infrastructure. Examples include home-area wireless networking [5], on-the-fly conferencing applications, networking intelligent devices or sensors, communication between mobile robots, etc.

Design of efficient routing protocols is the central challenge in such dynamic wireless networks. Much work has been done in this area starting from the seventies, when the U.S. Defense Research Agency, DARPA, supported the PR-

NET (Packet radio Network) [7] and SURAN (Survivable Adaptive Networks) [12] projects. They supported automatic route set up and maintenance in a packet radio network with moderate mobility. Interest in such networks has recently grown due to the common availability of wireless communication devices that can connect laptops and palmtops and operate in license free radio frequency bands (such as the Industrial-Scientific-Military or ISM band in the U.S.). In an interest to run internetworking protocols on ad hoc networks, a working group for Mobile, Ad hoc Networking (MANET) [9] has been formed within the Internet Engineering Task Force (IETF), whose charter includes developing a routing framework for running IP based protocols in ad-hoc networks. Several new routing protocols have been proposed in connection with the MANET working group efforts [9]. Of particular interest is the new class of *on-demand, source-initiated* protocols, that set up and maintain routes from a source to a destination on an “as needed” basis [6, 11]. This approach is in sharp contrast with the traditional *shortest path*-based protocols (e.g., link-state and distance vector [8]) that have been used successfully for a long time in dynamic, wireline networks, including the Internet.

1.1 On-demand Protocols and Multipathing

The motivation behind the on-demand protocols is that the “routing overhead” (typically measured in terms of the number of routing packets transmitted, as opposed to data packets) is typically lower than the shortest path protocols, as only the actively used routes are maintained. However, as some recent performance evaluation work has shown [4], the routing overhead still approaches to that of the shortest path protocols, if a moderate to large number of routes needs to be actively maintained (when, for example, there is a moderate to large number of active peer-to-peer conversations). This is because the on-demand protocols discover routes via a *flooding* technique, where the source (or any node seeking the route) floods the entire network with a query packet in search of a route to the destination. Flooding is also necessary for route maintenance activities, when a new route is needed, as the old one breaks because of node mobility. Flooding takes up a substantial amount of network bandwidth, which is at a premium in wireless networks. Efficient control of frequent network-wide flooding is thus important for the efficient performance of on-demand protocols. Some of our prior work was directed to limit the flood within a small region of the network [3] to reduce its impact on the network performance. In this paper, we focus on reducing the frequency of floods,

*This work is partially supported by AFOSR grant no. F49260-96-1-0472, Texas Advanced Technology Program grant no. 010115-248b, NSF MII grant no. CDA-9633299 and NSF CAREER award no. ACI-9733836.

by exploring multiple possible routes from a single flooded query. The goal is to provide enough redundancy at a low cost.

The idea of multipath routing is not new. It always has been a favorable alternative both for circuit switched and packet switched networks, as it provides an easy mechanism to distribute traffic and balance the network load, as well as provide fault tolerance. See, for example, [13] for some prior work on multipath routing on packet switched networks. Ad hoc network community also investigated multipath techniques, *albeit* less vigorously. The Temporally Ordered Routing Algorithm or TORA [10] provides multiple alternate paths by maintaining a “destination-oriented” directed acyclic graph (DAG) from the source. However, TORA does not have any easy mechanism to evaluate the “quality” of these multiple routes. Because of the nature of the protocol, it is hard to determine which route is the shortest. Also, TORA did not perform well in comparison to other on-demand protocols in some of the recent simulation studies [1, 4], ostensibly because the overhead incurred in maintaining multiple routes overwhelmed their performance benefits. The Dynamic Source Routing (DSR) protocol [2, 6] also has an option of maintaining multiple routes, so that an alternate route can be used upon failure of the primary one. In DSR, the quality of routes (i.e., hop-wise lengths) can be easily evaluated and the best (i.e., the shortest) one can be used. But in DSR [2] too many routes are maintained in a trivial manner, without any regard to their ultimate usefulness. In any case, performance benefits of multiple paths have never been evaluated.

Our goal in this paper is two-fold. First we propose an intelligent multipath technique for the DSR protocol that uses disjoint paths. This technique is also useful for any on-demand protocol using source-routed routing packets. Second, we develop an analytical modeling framework to evaluate the performance advantage of such multipath techniques. The modeling framework is also useful for performance evaluation of on-demand routing protocols regardless of the use of multiple paths. In our knowledge, this is the first attempt to analytically model on-demand routing protocols in ad hoc networks. Prior evaluations are based solely on simulation studies [1, 4].

The rest of the paper is organized as follows. In the next section, we briefly review the DSR protocol, which we use as the base protocol for our multipath techniques. We then present two multipath extensions for DSR. In Section 3, we develop analytical models to demonstrate how frequency of query floods are reduced with our multipath extensions. In Section 4, we present some numerical results obtained from the analytic models. We conclude in Section 5.

2 MULTIPATH DYNAMIC SOURCE ROUTING

The Dynamic Source Routing (DSR) protocol [2, 6] uses source routing – a technique where the source of a data packet determines the complete sequence of nodes through which to forward the packet; the source explicitly lists this route in the packet’s header. DSR builds routes on demand using a technique called *route discovery*. A source initiates

route discovery by flooding the network using query messages seeking a route to the destination. Each *query* (or *request*) message carries the sequence of hops it passed through in the message header. Once a query reaches the destination, the destination replies with a *reply* packet that simply copies the route from the query packet and traverses it backwards. Each node maintains a *route cache*, where complete routes to desired destinations are stored as learned from the reply packets. These routes are used by data packets. Route failure is detected by the failure of an attempted message transmission. Such a failure initiates an error packet to be sent backward to the source. The error packet erases all routes in the route caches of all intermediate nodes on its path, if the route contains the failed link. If a route is still needed a fresh route discovery is initiated.

DSR has a unique advantage by virtue of source routing. As the route is part of the packet itself, routing loops, either short- or long-lived, cannot be formed as they can be immediately detected and eliminated. This property opens up the protocol to a variety of useful optimizations. For example, a flooded query can be quenched early by having any non-destination host reply to the query if that host has a route to the intended destination. Ordinarily, this might cause routing loops, which would require elaborate mechanisms to detect or prevent. Also, a node can learn a route to a destination while passing on route reply packets. Finally, routes can be improved by having nodes promiscuously listen to conversations between other nodes in proximity.

2.1 Multipath Extension to DSR

In the proposed multipath extension, the destination replies to only a selected set of queries. Note that many copies of the flooded query message arrive at the destination via different routes. The queries that are replied to are those that carry a source route that is *link-wise disjoint* from the *primary* source route. Primary source route is the route taken by the first query reaching the destination. This usually defines the shortest route between the source and the destination. The destination “remembers” the primary route, in order to figure out which later requests to respond to. Only disjoint routes are chosen, as then a link failure in one route does not affect the others. This also implicitly controls the total number of replies, thus preventing a reply flood. The source keeps all routes received on reply packets in its route cache. When the primary route breaks, the shortest remaining *alternate* route is used. This process continues until all routes break, when a fresh route discovery is initiated. The number of alternate routes used can be a selectable parameter of the protocol. We will later show that only a few routes are actually sufficient. Let us call this technique as protocol 1, for ease of later reference.

Protocol 1 equips only the source with alternate routes. An intermediate link failure on the primary route still sends an error packet back to the source, which will then use an alternate route. This causes a temporary loss of route for the data packets that are already in transit upstream from the failed link. To prevent this, a better alternative is explored. All intermediate nodes are now equipped with a disjoint, alternate route so that in-transit data packets no longer face any

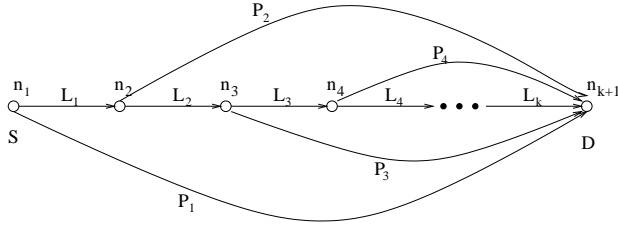


Figure 1: Multipath protocol 2. The primary route is depicted by the sequence of links $L_1 - L_2 - \dots - L_k$. Each node in the primary route n_i , has an alternate path P_i to the destination.

route loss. To accomplish this, the destination now replies to *each* intermediate node in the primary route with an alternate disjoint route to the destination. Note that any such reply is in response to a query from the source that has traveled through that intermediate node. The reply is targeted to the intermediate node instead of the source. See Figure 1 for an illustration of how the routes are maintained. Note that it may not always be possible for all intermediate nodes to get an alternate disjoint route. This would be particularly true for sparse networks. Thus, this still may result in temporary loss of routes on link failures until an upstream node switches to an alternate route. For the sake of simplicity of modeling, we will ignore this possibility in our later evaluations.

Refer to Figure 1 to see the utility of the multiple routes maintained in each intermediate nodes. The source S uses the primary route for transmitting data packets to D until it breaks at some point, say L_i . When the link L_i is broken, the node n_i responds to the situation by replacing the unused portion of the route, $L_i - L_k$, in the data packet header by the alternate route P_i . This will continue until a link on P_i breaks. It will cause an error packet transmitted backward up to node n_{i-1} , which will quench the error packet and switch all later data packets to its own alternate route P_{i-1} by modifying the source route in the packet header as before.

Thus, loss of *all* routes in a node to the destination generates an error packet back to the source. Any intermediate node with an alternate path to the destination quenches the error packet. This node is also responsible for modifying the source route on all later data packets to use its own alternate route. This process continues until the source gets an error packet and has no alternate route to fall back on. Then a new route discovery is initiated. We refer to this technique as protocol 2. Note that unlike protocol 1, we maintain only one alternate path per intermediate node in protocol 2. In principle, any number of such paths can be maintained. But as we will show in our modeling work later, more than one alternate path provides only a very minimal additional advantage.

3 ANALYTIC MODELING

We represent the lifetime of a wireless link between a pair of nodes by a random variable. Consider a route from S to D that consists of a sequence of k wireless links over $k - 1$ intermediate nodes. Let L_i be the i -th link in the route. The lifetime of L_i is denoted by X_{L_i} . Assume that X_{L_i} , $i = 1, 2, \dots, k$, are independent and identically distributed (iid)

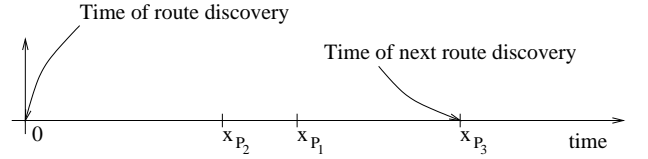


Figure 2: Example of multipath routing where the source has three independent routes P_1, P_2, P_3 to the destination ($N=3$). The figure represents the lifetimes of the three routes. When the primary route P_1 breaks at time x_{P_1} , S attempts to use P_3 as P_2 is already broken. P_3 breaks at time x_{P_3} , when new route discovery is initiated.

exponential random variables, each with mean l . We will use upper-case alphabets to denote random variables and the corresponding lower-case alphabets to denote their values.

Since a route fails when *any one* of the wireless links in its path breaks, the lifetime of a route P , consisting of k wireless links, is a random variable X_P that can be expressed as

$$X_P = \min(X_{L_1}, X_{L_2}, \dots, X_{L_k}) \quad (1)$$

It is well known that X_P is also an exponentially distributed random variable with a mean of $\frac{l}{k}$ [14].

Using the basic assumptions about the link failure behavior as above, we proceed to derive the statistics of the time interval between successive route discoveries for the proposed multipath techniques.

3.1 Modeling of Protocol 1

Assume a source S has N routes to a destination D . The primary route is denoted by P_1 and the $N - 1$ alternate routes are denoted by P_2, P_3, \dots, P_N . The length of route P_i is k_i . See Figure 2. The time after which none of the routes are useful is a random variable T , where

$$T = \max(X_{P_1}, X_{P_2}, \dots, X_{P_N}) \quad (2)$$

T represents the time between successive route discoveries. Here, we assume that the end-to-end packet transmission latency in the network is very small compared to the interval between route changes. Thus, the times to discover routes or propagate error packets etc. can be ignored relative to T . These are very reasonable assumptions, as otherwise routes will fail while discovery or repair is in progress. No routing protocol will perform well in such dynamic conditions.

Claim: The probability density function (pdf) of T , the time between successive route discoveries, is given by

$$\begin{aligned} f_T(t) = & \lambda_1 e^{-\lambda_1 t} (1 - e^{-\lambda_2 t}) (1 - e^{-\lambda_3 t}) \dots (1 - e^{-\lambda_N t}) \\ & + \lambda_2 e^{-\lambda_2 t} (1 - e^{-\lambda_1 t}) (1 - e^{-\lambda_3 t}) \dots (1 - e^{-\lambda_N t}) + \dots \\ & + \lambda_N e^{-\lambda_N t} (1 - e^{-\lambda_1 t}) (1 - e^{-\lambda_3 t}) \dots (1 - e^{-\lambda_{N-1} t}) \end{aligned} \quad (3)$$

where $\lambda_i = k_i/l = 1/\text{lifetime of the } i\text{-th route}$.

Proof: Consider N iid exponential random variables, $X_{P_1}, X_{P_2}, \dots, X_{P_N}$, where the pdf of X_{P_i} is $f_{X_{P_i}}(t) =$

$\lambda_i e^{-\lambda_i t}$, $i = 1, 2, \dots, N$. Note that X_{P_i} 's are independent. Therefore, the cumulative distribution function (cdf) of T , $F_T(t)$, is obtained as

$$\begin{aligned} F_T(t) &= P[T \leq t] \\ &= P[\max(X_{P_1}, X_{P_2}, \dots, X_{P_N}) \leq t] \\ &= P[(X_{P_1} \leq t) \cap (X_{P_2} \leq t) \cap \dots \cap (X_{P_N} \leq t)] \\ &= \prod_{i=1}^N F_{X_{P_i}}(t) \end{aligned} \quad (4)$$

where $F_{X_{P_i}}(t) = 1 - e^{-\lambda_i t}$ is the cdf of X_{P_i} . By differentiating (4) with respect to t , we get the pdf of T as shown in (3). \square

The expected value of T can be derived from (3) by knowing the hop-wise lengths of all the routes, k_i , $i = 1, 2, \dots, N$. For example, for $N = 2$, the expected value of T , $E[T]$ is

$$E[T] = \frac{\lambda_1^2 + \lambda_2^2 + \lambda_1 \lambda_2}{\lambda_1 \lambda_2 (\lambda_1 + \lambda_2)} \quad (5)$$

For the case that the primary route has $k_1 = 3$ hops, and the alternate route has $k_2 = 4$ hops, we have $E[T] = \frac{9+16+12}{12 \times 7} l = \frac{37}{84} l$. Compare this with DSR with only a single route of 3 hops, where $E[T] = \frac{l}{3}$. This represents almost 25% reduction in the frequency of route discoveries compared to the single path case.

Consider also the special case when all the routes are equal, i.e. $k_1 = k_2 = \dots = k_N = k$. In this case,

$$\begin{aligned} E[T] &= \frac{N}{\lambda} - \frac{N}{2^2 \lambda} \binom{N-1}{1} + \frac{N}{3^2 \lambda} \binom{N-1}{2} + \\ &\dots + (-1)^{N-1} \frac{1}{N \lambda} \end{aligned} \quad (6)$$

where $\lambda = k/l$.

3.2 Modeling of Protocol 2

The analysis in this case is a little more involved, as a new route discovery will be initiated only when all intermediate nodes lose their downstream links as well as the alternate routes. Let \overline{L}_i denote the event that the link L_i fails, and \overline{P}_i denote the event that the path P_i fails (see Figure 1). Then the time until the next route discovery T , is the time until the event E is true, where E is described by the following logical expression:

$$\begin{aligned} E &= (\overline{L}_1 \overline{P}_1) + (\overline{L}_2 (\overline{L}_1 + \overline{P}_2) \overline{P}_1) + (\overline{L}_3 (\overline{L}_1 \\ &+ (\overline{L}_2 + \overline{P}_3) \overline{P}_2) \overline{P}_1) + \dots \end{aligned} \quad (7)$$

The i^{th} term inside the braces of the right-hand side of (7) represents the events starting with the failure of L_i and leading to a new route discovery. For example, the second term represents the following sequence of events:

- L_2 breaks on the primary route, prompting S to form the alternate route $L_1 - P_2$.
- The alternate route breaks when either L_1 or P_2 breaks, prompting S to use the new alternate route P_1 .
- This route fails when P_1 breaks, causing S to start a new route discovery

Hence, starting with the breakage of L_2 , the events leading to a new route discovery from S is $\overline{L}_2 (\overline{L}_1 + \overline{P}_2) \overline{P}_1$. The other terms can be derived similarly.

Equation 7 can be simplified to a more compact form:

$$\begin{aligned} E &= \overline{L}_1 \overline{P}_1 + \overline{L}_2 \overline{P}_2 \overline{P}_1 + \overline{L}_3 \overline{P}_3 \overline{P}_2 \overline{P}_1 + \dots \\ &+ (\overline{L}_k \overline{P}_k \overline{P}_{k-1} \dots \overline{P}_1) \end{aligned} \quad (8)$$

which is also quite intuitive. Then T can be expressed as

$$\begin{aligned} T &= \min(\max(X_{L_1}, X_{P_1}), \max(X_{L_2}, X_{P_2}, X_{P_1}), \\ &\dots, \max(X_{L_k}, X_{P_k}, X_{P_{k-1}}, \dots, X_{P_1})) \end{aligned} \quad (9)$$

Recall that X_{L_i} is an exponential random variable with mean l . If the alternate path P_i consists of k_i hops, then X_{P_i} is also an exponential random variable, with mean l/k_i . Note again that according to our assumptions, X_{L_i} and X_{P_i} are independent.

Let us denote the random variable $\max(X_{L_i}, X_{P_i}, X_{P_{i-1}}, \dots, X_{P_1})$ by Z_i . Hence, following (3), the pdf of Z_i , $f_{Z_i}(t)$ is given by

$$f_{Z_i}(t) = \sum_{j=1}^{i+1} \left(\lambda_j^{(i)} e^{-\lambda_j^{(i)} t} \prod_{k=1, k \neq j}^{i+1} (1 - e^{-\lambda_k^{(i)} t}) \right), \quad (10)$$

where

$$\lambda_j^{(i)} = \begin{cases} \frac{k_j}{l} & \text{for } j = 1, 2, \dots, i \\ \frac{1}{l} & \text{for } j = i+1 \end{cases} \quad (11)$$

We now derive the pdf of $T = \min(Z_1, Z_2, \dots, Z_k)$.

Claim: The pdf of T is given by

$$f_T(t) = \sum_{i=1}^k \left(f_{Z_i}(t) \prod_{j=1, j \neq i}^k (1 - F_{Z_j}(t)) \right) \quad (12)$$

Proof: From definition, the pdf of T can be written as

$$f_T(t) = \lim_{dt \rightarrow 0} \frac{P[t \leq T \leq t + dt]}{dt} \quad (13)$$

Consider,

$$\begin{aligned} &P[t \leq T \leq t + dt] \\ &= P[t \leq \min(Z_1, Z_2, \dots, Z_k) \leq t + dt] \end{aligned}$$

$$\begin{aligned}
&= \sum_{i=1}^k P[t \leq Z_i \leq t + dt | Z_i = \min(Z_1, Z_2, \dots, Z_k)] \\
&\quad \cdot P[Z_i = \min(Z_1, Z_2, \dots, Z_k)] \\
&= \sum_{i=1}^k P[t \leq Z_i \leq t + dt] \\
&\quad \cdot P[Z_i = \min(Z_1, Z_2, \dots, Z_k)] \quad (14)
\end{aligned}$$

Dividing both sides by dt and taking the limit as dt approaches zero, we get the desired result:

$$\begin{aligned}
f_T(t) &= \sum_{i=1}^k f_{Z_i}(t) \prod_{j=1, j \neq i}^k P[Z_j > Z_i] \\
&= \sum_{i=1}^k f_{Z_i}(t) \prod_{j=1, j \neq i}^k (1 - F_{Z_j}(t)) \quad (15)
\end{aligned}$$

□

4 NUMERICAL RESULTS

In this section we present some numerical results showing the performance benefits of the multipath routing protocols using the analysis presented previously. Let us start with the protocol 1. The pdf of the time interval between successive route discoveries is given by Equation 3. We evaluate the expected value of this interval using numerical techniques for three special cases. In the first case (*Case A*), we assume that all the N paths from S to D are of the same length (Equation 6). This implies the “best case” scenario for multipath routing, where N disjoint shortest path routes exist between the given source–destination pair. In the second (*Case B*) and third (*Case C*) cases, we assume that the N disjoint routes between the source and destination are of increasing lengths, with the primary route being the shortest. For *Case B* the successive path lengths increase by one, and for *Case C* they increase by two. Thus,

$$\begin{aligned}
\text{Case A:} \quad &k_1 = k_2 = \dots = k_N = k \\
\text{Case B:} \quad &k_1 = k, k_2 = k + 1, \dots, k_N = k + N - 1 \\
\text{Case C:} \quad &k_1 = k, k_2 = k + 2, \dots, k_N = k + 2(N - 1)
\end{aligned}$$

The expected time interval between successive route discoveries at source for the three cases are plotted against different values of the path length of the primary route, in Figure 3. The mean lifetime of a single link (l) is assumed to be 5, for this plot and all later plots. The interval increases from Cases *C* to *B* to *A*. For all cases the interval is longer than the single path case denoting less frequent route discovery. Also, relative advantage of multipath routing diminishes as the primary route gets longer. This is intuitive, as the alternate routes are at least as long, and longer routes typically break earlier.

In Figure 4 the expected interval between route discoveries in protocol 1 is plotted against the number of routes (N) maintained, all for the same primary route length. Two pri-

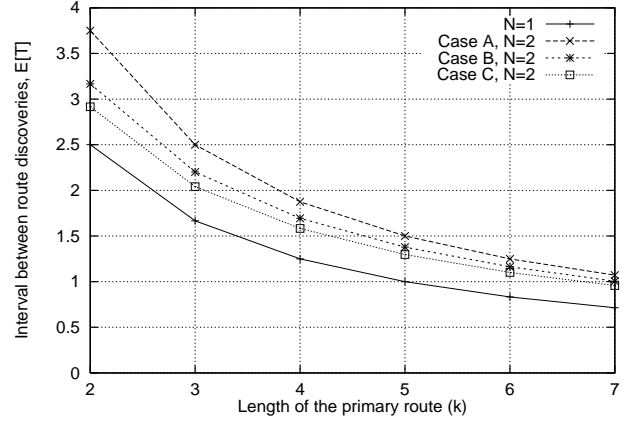


Figure 3: Performance of protocol 1 with different lengths of the primary route. Three cases for two routes (one primary, and one alternate) are compared with the single path case. Mean lifetime of a single link = 5.

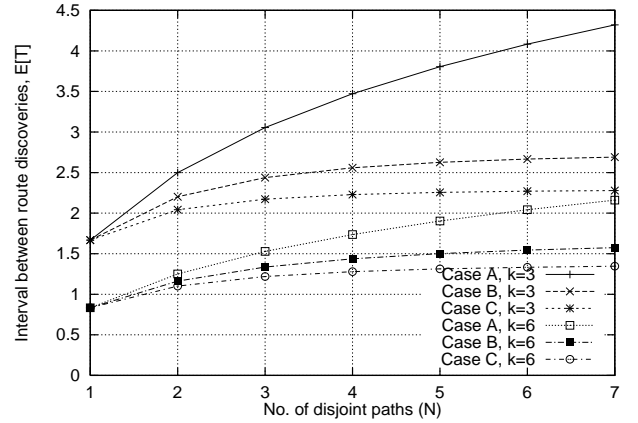


Figure 4: Performance of protocol 1 with varying number of paths (N). Two different lengths of the primary path are used, $k = 3$ and 6.

mary route lengths are used ($k = 3$ and 6). As expected performance always improves with increasing number of alternate routes. However, the incremental improvement is very small for $N > 3$, except when the paths are of the same length (Case A), Note that this case is very unlikely to occur in practice. This indicates that usually only one or two additional routes will be sufficient.

In protocol 2 the performance is dependent on the values of k_i , $i = 1, 2, \dots, k$, i.e. the path lengths in the alternate routes from each intermediate node. The actual values of these parameters are dependent on the dynamic conditions of the network. To get an idea of the performance improvement with multipath routing, we consider three different estimates of these parameters. In case A, we assume that the alternate routes from each intermediate node n_i are of the same length as that of the primary route from n_i to the destination. In cases B and C, we assume that the alternate route is

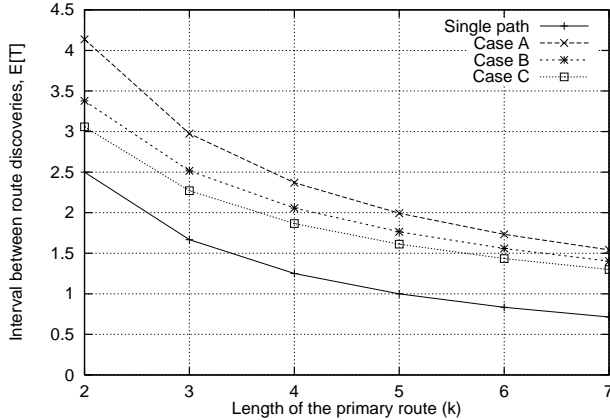


Figure 5: Performance of multipath routing protocol 2 with different lengths of the primary route. Performance of the single path routing is shown as a reference.

larger than the primary route by one and two, respectively. Note that we consider only a single alternate route per node in protocol 2. Since the primary route is of length k , the path lengths of P_i , $i = 1, 2, \dots, k$, are:

$$\text{Case A : } k_i = k - i + 1$$

$$\text{Case B : } k_i = k - i + 2$$

$$\text{Case C : } k_i = k - i + 3$$

For these three cases, we again use numerical techniques to determine the expected time interval between route discoveries ($E[T]$) from the pdf of T as given by equation 12. We plot the expected time interval for some special cases in Figure 5. As expected, the protocol 2 performs significantly better than single path routing.

To compare the relative performance of protocols 1 and 2, we look at *Case A* for both protocols, where the “best” alternate routes are assumed to be available for both protocols. The expected time between successive route discoveries are compared in Table 4. Also shown in Table 4 are frequencies of route discoveries for multipath routing protocols relative to the single path routing. As expected protocol 2 provides more substantial performance advantage.

Table 1: Comparison between the two variations of the multipath protocols. All frequencies are relative to single path routing.

k	Protocol 1 (N=2)		Protocol 2	
	$E[T]$	Frequency of route discovery	$E[T]$	Frequency of route discovery
3	2.5	0.668	2.67	0.560
4	1.875	0.667	2.03	0.527
5	1.50	0.667	1.64	0.502
6	1.25	0.667	1.37	0.481
7	1.07	0.667	1.18	0.462

5 CONCLUSIONS

We proposed a multipath extension for the popular on-demand routing protocol DSR. The extension explores alternate, disjoint routes that can be useful in case the primary route breaks. Two variations are explored. In the first, only the source gets multiple alternate routes. In the second, each intermediate node on the primary route gets an alternate route. The key advantage is the reduction in the frequency of route discovery flood, which is recognized as a major overhead in on-demand protocols. We also provide a framework for modeling the time interval between successive route discoveries for on-demand protocols based on simple assumptions on the lifetime of a single wireless link. Evaluation of the multipath routing extension in this framework demonstrates that there are definite advantages to be gained from providing alternate routes on intermediate nodes, in addition to the source node. In any case, any form of multipath technique always performs substantially better than single path routing. The modeling effort also shows that longer alternate paths are less advantageous, as they tend to break too early. This indicates that it is only useful to explore alternate routes with some bound on the hop-wise path length. Also, the performance advantage from using more than two alternate routes is usually minimal.

Multipath techniques are not without pitfalls, however. It is worthwhile to mention them clearly, as they are not captured in our analytical model. Alternate routes in practice will always tend to be longer than the primary routes, especially when we focus only on disjoint routes. This will make the end-to-end delay of data packets longer, presenting a trade-off between end-to-end delay and routing overhead. In addition, maintenance cost of multiple routes in terms of additional routing packets should be evaluated. The right choice will depend on the actual load on the network and the requirements of the application. Our future work will extend the model to capture this tradeoff and will use simulation studies to validate and refine the model.

REFERENCES

- [1] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” *Proceedings of the 4th International Conference on Mobile Computing and Networking (ACM MOBICOM’98)*, pages 85–97, October 1998.
- [2] Josh Broch, David Johnson, and David Maltz, “The dynamic source routing protocol for mobile ad hoc networks,” <http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-01.txt>, Dec 1998. IETF Internet Draft.
- [3] R. Castaneda and S. R. Das, “Query localization techniques for on-demand protocols in ad hoc networks,” *Proc. ACM Mobicom Conference*, August 1999, to appear.
- [4] S.R. Das, R. Castaneda, J. Yan, and R. Sengupta, “Comparative performance evaluation of routing protocols for mobile, ad hoc networks,” *7th Int. Conf.*

on *Computer Communications and Networks (IC3N)*, pages 153–161, October 1998.

- [5] K. J. Negus et. al., “HomeRF and SWAP: Wireless networking for the connected home,” *ACM SIGMOBILE Mobile Computing and Communications Review*, 2(4):28–37, Oct 1998.
- [6] D. Johnson and D. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile computing*, T. Imielinski and H. Korth, editors, Kluwer Academic, 1996.
- [7] John Jubin and Janet D. Tornow, “The DARPA packet radio network protocols,” *Proceedings of the IEEE*, 75(1):21–32, January 1987.
- [8] S. Keshav, *An Engineering Approach to Computer Networking : ATM Networks, the Internet, and the Telephone Network*, chapter 11. Addison-Wesley, 1997.
- [9] J. Macker and S. Corson, “Mobile ad hoc networks (MANET),” <http://www.ietf.org/html.charters/manet-charter.html>, 1997, IETF Working Group Charter.
- [10] V. D. Park and M. S. Corson, “A highly adaptive distributed routing algorithm for mobile wireless networks,” *Proceedings of IEEE INFOCOM’97 Conf.*, April 1997.
- [11] Charles Perkins and Elizabeth Royer, “Ad hoc on-demand distance vector routing,” *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, Feb 1999.
- [12] N. Schacham and J. Westcott, “Future directions in packet radio architectures and protocols,” *Proceedings of the IEEE*, 75(1):83–99, Jan 1987.
- [13] W.T. Zaumen and J. J. Garcia-Luna-Aceves, “Shortest multipath routing using generalized diffusing computations,” *Proceedings IEEE INFOCOM 98*, March 1998.
- [14] S. Ross, *Introduction to Probability Models*, Academic Press, 1989.