Routing Reliability Analysis of Segmented Backup Paths in Mobile Ad Hoc Networks

Ajay Agarwal and Bijendra Jain
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
Hauz Khas New Delhi 110016, INDIA
Email: {aagar, bnj}@cse.iitd.ernet.in

ABSTRACT

Several real-time applications (e.g., video conferencing, remote control systems) demand guarantees on the message delivery latency. Supporting such QoS constrained connections requires the existence of a routing mechanism, which computes paths that satisfy QoS constraints In mobile ad hoc network, wireless links tend to frequently fail as nodes move in and out of transmission range of one another. Providing fault tolerance with QoS guarantees in such networks is challenging. An algorithm called Segmented Backup Source Routing (SBSR) provides fault tolerance with end-to-end delay as the QoS parameter. The algorithm constructs a *set* of delay-constrained segmented backup paths. Each such segmented backup path, protects a segment of the primary path rather than the entire path. This approach has two advantages. First, one is able to identify backup paths for *any* selected primary path, as long as there exists a pair of node disjoint paths from source to destination (In other words, if there are two node disjoint paths, one is not required to use either of them as the primary path). This has the advantage that the primary path may be selected based on QoS considerations rather than a consideration of fault tolerance. The second significant advantage is that the connection reliability of the segmented Backup path set is higher than the two disjoint paths. In this paper, we design a framework to analyse the connection reliability of such a segmented backup path set.

I. INTRODUCTION

There is growing interest among mobile users to access a variety of applications such as real-time distributed computation and video conferencing. Such distributed real-time applications demand guarantees on QoS parameters, such as end-to-end delay, jitter and packet loss. QoS requirements are agreed upon before identifying the data path. Further, it is crucial that the QoS constraints continue to ~~met~~ even when one or more nodes or links fail.

Since nodes in a mobile ad hoc network move freely and without constraints, routes often get broken. The main challenge is to then ensure that routing protocols respond to failure of one or more nodes or links in a timely manner, by reconstructing the routes such that QoS requirements continue to be met.

A simple and effective method to provide QoS guarantees in the face of node or link failure is to use an alternative path which is disjoint with the primary path (see [1], [2], [3], [4], [5]). However, this strategy based on end-to-end backup has several drawbacks. These are discussed below.

1) Clearly, an end-to-end backup path should be node disjoint with the primary path. However, it is usually the case that the primary path is determined based on QoS constraints first, and then a node disjoint path is identified as the backup path. In some cases, the primary path so determined may be routed through the network in a manner such that it "blocks" all node-disjoint paths in the network. In Figure 1, for example, there exist two node disjoint paths $B1 = \langle S,B,D \rangle$ and $B2 = \langle S,C,D \rangle$ from source S to destination D (shown by dotted lines). However, if the primary path selected is $P = \langle S,B,C,D \rangle$, then there is no path which is node disjoint with this primary path.

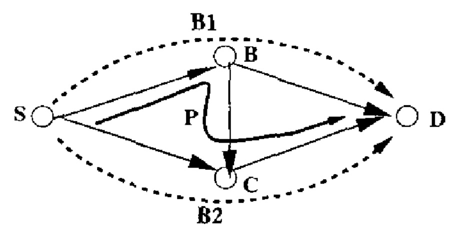

Fig. 1. No end-to-end backup path, if primary path is $\langle S,B,C,D \rangle$.

2) Both, the primary and its backup path, must separately satisfy all QoS requirements. This is a very difficult constraint to satisfy since the number of end-to-end

node disjoint paths, if available, is small.

In this paper, we analyse the connection reliability of a *segmented backup path set* (see [6] for details). A segmented backup path *set* consists of multiple segmented backup paths, each of which is from the given source to the destination, and is the path to be used in case one of the nodes on the primary path fails.

In Figure 2, for example, the segmented backup path *set* consists of three segmented backup paths, viz., $\pi_1 = \langle S,A,B,4,5,6,7,8,D\rangle$, $\pi_2 = \langle S,2,3,C,E,7,8,D\rangle$ and $\pi_3 = \langle S,2,3,4,5,6,F,D\rangle$. Path π_1 is used when either node 2 or 3 fails. Similarly, π_2 is used to backup failure of nodes 4, 5 or 6, while π_3 is used when node 7 or 8 fails. This is unlike an

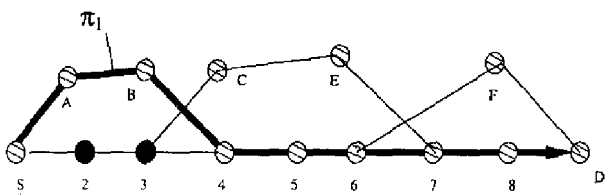

Fig. 2. A segmented backup path π_1 (shown bold) is used when either node 2 or 3 fails.

end-to-end scheme where a backup path provides a backup in case any node on the primary path fails. This new scheme based on backup paths has the following advantages over a scheme that is based on end-to-end backup

1) It can be shown that for any given primary path between a given pair of source and destination nodes, S and D, there exists at least one segmented backup path set which together provide protection against failure of any intermediate node on the primary path, provided there exists a pair of node disjoint paths, between S and D. The significance of this result is that one is free to select any primary path which satisfies the given QoS constraints independent of whether there exist a path which is node disjoint to the particular primary path.
2) In fact, for a given primary path, one would be able to discover at least one set of segmented backup paths. This offers greater opportunity to ensure that a backup path also meets QoS requirements.
3) A third advantage of using segmented backup path is that the time it takes to detect and notify failures, and to switch traffic from the primary to a segmented backup path, can be (and will be) smaller. It is because a segmented backup path set consists of multiple backups, each of which spans only a part of the primary path. This enables the network to recover from a failure by simply activating a *local* bridge, rather than switching to a completely new end-to-end path ([7], [8], [9], [10], [11]).

The rest of the paper is organized as follows. In Section II, we explain the concept of segmented backup path set, and illustrate their advantages over end-to-end backups. Section III illustrates the analysis of overall connection reliability. Concluding remarks are made in Section IV.

II. SEGMENTED BACKUP PATHS

In this section, we explain in detail the scheme based on segmented backup paths. In our scheme, we find backups for the primary path, taken in parts. We do not consider link failures explicitly but these are effectively taken care of by node failures.

A. Formal Definitions

Segmented Backup Path:

Consider a network G = (N, L), where $N = \{1, 2, \ldots, |N|\}$ is the set of nodes, and $L = \{\langle i, j\rangle | i \neq j, i, j \in N\}$ is the set of bidirectional links. Further, for a given source node, $S = p_1$ and destination node, $D = p_n$, let $P = \langle p_1, p_2, \ldots, p_{n-1}, p_n\rangle$ or $\langle S, p_2, \ldots, p_{n-1}, D\rangle$, be a primary path. We define a *segment* $\sigma = \{p_i, p_{i+1}, \ldots, p_j\}$, where $i > 1$ and $j < n$, to be a set of contiguous nodes along the primary path. Then, a sub-path $B(\sigma)=\langle x_1, x_2, \ldots, x_k\rangle$ is said to be a *bridge* across the segment σ, if $x_1 = p_{i-1}$, $x_k = p_m$, $j+1 \leq m \leq n$ provided the sub-path $\langle x_2, x_3, \ldots, x_{k-1}\rangle$ and the primary path P are node-disjoint. The corresponding *segmented backup path* is $\pi(\sigma) = \langle S, p_2, \ldots, p_{i-2}, x_1, x_2, \ldots, x_{k-1}, x_k, p_{m+1}, \ldots, D\rangle$. The path $\pi(\sigma)$ has the property that it may be used to transfer packets from S to D in case any node $p \in \sigma$ fails.

The first node x_1 of the bridge $B(\sigma)=\langle x_1, x_2, \ldots, x_k\rangle$, is said to be a *Segment Switching Router* (SSR), and its last node x_k is referred to as *Segment Merging Router* (SMR). In Figure 2, for instance, if a node p $\in$ {4,5,6} fails, then the SSR (node 3) re-routes data packets along $\langle 3, C, E, 7\rangle$, while corresponding SMR (node 7) forwards the packets along the primary path, again.

Further, in order to avoid infinite looping of data packets, we insist that for any two successive segments $\sigma_t = \langle p_i, p_{i+1}, \ldots, p_j\rangle$ and $\sigma_{t+1} = \langle p_{j+1}, p_{j+2}, \ldots, p_k\rangle$, the corresponding bridges $B(\sigma_t) = \langle x_1, \ldots, x_p\rangle$, $x_1 = p_{i-1}$, $x_p = p_u$ (for some u, $j+1 \leq u \leq n$) and $B(\sigma_{t+1}) = \langle y_1, \ldots, y_q\rangle$, $y_1 = p_j$, $y_q = p_v$ (for some v, $k+1 \leq v \leq n$) are such that (see Figure 3)

$$i \leq j < u \leq v. \quad (1)$$

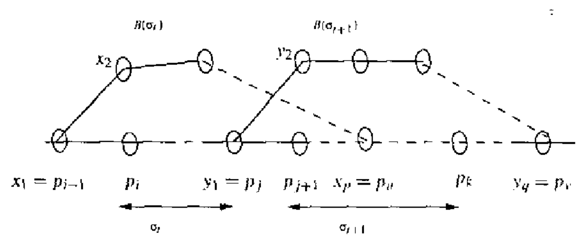

Fig. 3. Successive bridges

Segmented Backup Path Set:

We now define a *segmented backup path set* in order to ensure that an alternative path is available in case any node along a primary path fails. For the given primary path $P = \langle S, p_2, \ldots, p_{n-1}, D\rangle$, we consider the *ordered* collection of pairwise disjoint segments of P, viz., $\{\sigma_1,\ \sigma_2, \ldots\ \sigma_m\}$, where $m \geq 0$ and each segment

$$\sigma_r = \{p_i^r,\ p_{i+1}^r, \ldots,\ p_j^r\},\ 1 \leq r \leq m \tag{2}$$

where, $p_j^r = p_k$ implies $p_i^{r+1} = p_{k+1}$, for $r = 1, 2, \ldots, m-1$, $p_i^1 = S$, $p_j^m = D$.

This definition of segments ensures that every intermediate node on the primary path P is contained in some segment. In Figure 2, for example, $m = 3$ and $\sigma_1 = \{2,\ 3\}$, $\sigma_2 = \{4,\ 5,\ 6\}$, $\sigma_3 = \{7,\ 8\}$.

Now that we have a complete set of segments, we can define a corresponding segmented backup path *set* to be

$$\Pi(P) = \cup_{r=1,2,\ldots,m}\ \pi(\sigma_r) = \cup_{r=1,2,\ldots,m}\ \pi_r, \tag{3}$$

where $\pi(\sigma_r)$ (or π_r in short) is a segmented backup path corresponding to segment $\sigma_r = \{p_i^r, p_{i+1}^r, \ldots, p_j^r\}$. This path has the property that if a node $p_k^r \in \sigma_r$ were to fail then all traffic will be re-routed over the corresponding segmented backup path π_r. Further, the responsibility to switch the traffic will be that of the corresponding SSR, viz., p_j^{r-1}. (see Figure 4).

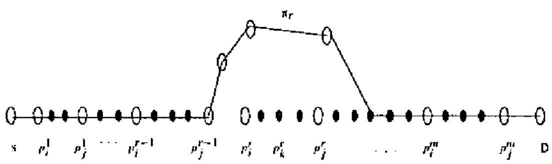

Fig. 4. Segmented backup path π_r.

From Figure 4, it should also be clear that the segmented backup path π_r may be used to protect against failure of certain nodes other than those in σ_r. In Figure 5, for example, the segmented backup paths are $\pi(\sigma_1) = \langle S, A, B, 4, 5, 6, 7, 8, D\rangle$, $\pi(\sigma_2) = \langle S, 2, 3, C, E, F, D\rangle$ and $\pi(\sigma_3) = \langle S, 2, 3, 4, 5, 6, F, D\rangle$. While nodes in σ_2 are protected against failure by $\pi(\sigma_2)$, nodes in σ_3 may be protected by $\pi(\sigma_2)$ *or* by $\pi(\sigma_3)$. In the proposed scheme we insist that if a node in σ_3 were to fail then the segmented backup path $\pi(\sigma_3)$ be used. That is, the responsibility of detecting failure and switching traffic to the bridge across σ_3 be with the closest segment switching router, viz., node 6 in the above example. This is desirable since the time to detect and notify failure is minimum for node 6 (as opposed to node 3).

Based on the discussion above, it becomes clear as to how one may obtain a set of segments from a given set of segmented backup paths, Π, corresponding to a given path P. To illustrate, let primary path $P = \langle S, p_2, \ldots, p_{n-1}, D\rangle$

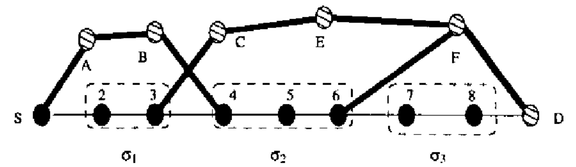

Fig. 5. Even if nodes 7 or 8 are protected by $\pi(\sigma_2)$, path $\pi(\sigma_3)$ is used to switch traffic if any one of them fails.

where $S = p_1$ and $D = p_n$, and $\Pi = \{\pi(\sigma_1),\ \pi(\sigma_2), \ldots \pi(\sigma_m)\}$. Further, let $\mathcal{N}(P)$ or $\mathcal{N}(\pi_m)$ denote the collection of nodes on a path P or π_m (respectively).

Then, clearly,

$\sigma_m = \mathcal{N}(P) - \mathcal{N}(\pi_m)$,

$\sigma_{m-1} = \mathcal{N}(P) - \mathcal{N}(\pi_{m-1}) - \sigma_m$,

...

...

...

$\sigma_1 = \mathcal{N}(P) - \mathcal{N}(\pi_1) - \sigma_m - \sigma_{m-1} - \ldots - \sigma_2$.

In Figure 5, for instance, m=3. Then,

$\sigma_3 = \{7, 8\}$,

$\sigma_2 = \{S, 2, 3, 4, 5, 6, 7, 8, D\} - \{S, 2, 3, C, E, F, D\} - \{7, 8\} = \{4, 5, 6\}$, and

$\sigma_1 = \{2, 3\}$.

We now illustrate the advantage that segmented backup paths offer, particularly in respect of ensuring that delay constraints are met even when a node fails. To start with, note that for a given path $P = \langle p_1,\ p_2,\ p_3, \ldots\ p_n\rangle$, the end-to-end delay is given by $\delta(P) = \sum_{i=1}^{n-1} delay(\langle p_i, p_{i+1}\rangle)$. Consider,

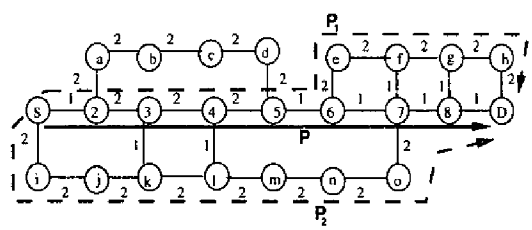

Fig. 6. Improved QoS guarantees.

for instance, the network given in Figure 6, wherein we have also indicated the delay for each link. There are two node-disjoint paths $P_1 = \langle S, 2, 3, 4, 5, 6, e, f, g, h, D\rangle$ and $P_2 = \langle S, i, j, k, l, m, n, o, 7, 8, D\rangle$ and the corresponding delays are $\delta(P_1) = \delta(P_2) = 18$. If the stated QoS constraint on end-to-end delay is 17 then clearly both P_1 and P_2 do not meet the requirement. If, however, the primary path is chosen to be $P = \langle S,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7,\ 8,\ D\rangle$ then there is no other path which is disjoint from this primary path, P. However, there may still be ways to identify and construct a segmented backup path set. For instance, if $\sigma_1 = \{2\}$, $\sigma_2 = \{3,\ 4\}$, $\sigma_3 = \{5,\ 6\}$, $\sigma_4 = \{7\}$ and $\sigma_5 = \{8\}$, then one set of corresponding segmented backup paths are $\pi(\sigma_1) = \langle S,i,j,k,3,4,5,6,7,8,D\rangle$, $\pi(\sigma_2) = \langle S,2,a,b,c,d,5,6,7,8,D\rangle$, $\pi(\sigma_3) = \langle S,2,3,4,l,m,n,o,7,8,D\rangle$, $\pi(\sigma_4) = \langle S,2,3,4,5,6,e,f,g,8,D\rangle$, and $\pi(\sigma_5) = \langle S,2,3,4,5,6,7,f,g,h,D\rangle$. The corresponding delay for these segmented backup paths are $\delta(\pi(\sigma_1)) = 15$, $\delta(\pi(\sigma_2)) = 15$, $\delta(\pi(\sigma_3)) = 16$, $\delta(\pi(\sigma_4)) = 16$, $\delta(\pi(\sigma_5)) = 16$. Clearly, all of these meet the required delay constraint. Alternatively, if we select $\sigma_1 = \{2, 3\}$, $\sigma_2 = \{4, 5, 6\}$, $\sigma_3 = \{7\}$, $\sigma_4 = \{8\}$, together with corresponding segmented backup paths $\pi(\sigma_1) = \langle S,i,j,k,l,4,5,6,7,8,D\rangle$, $\pi(\sigma_2) = \langle S,2,3,k,l,m,n,o,7,8,D\rangle$, $\pi(\sigma_3) = \langle S,2,3,4,5,6,e,f,g,8,D\rangle$, $\pi(\sigma_4) = \langle S,2,3,4,5,6,7,f,g,h,D\rangle$, then the delay for these segmented backup paths are $\delta(\pi(\sigma_1)) = 15$, $\delta(\pi(\sigma_2)) = 16$, $\delta(\pi(\sigma_3)) = 16$, $\delta(\pi(\sigma_4)) = 16$. These backup paths again meet the delay constraint.

An interesting and important point that needs to be made is that one can be sure that for any given primary path, P, between a source S and a destination D, one will be able to identify at least one set of segments and corresponding set of segmented backup paths, provided the network is such that there are two or more node disjoint paths between S and D. In other words, by working with segmented backup paths, instead, one can explore many more alternative paths to meet delay (and other QoS) constraint(s) when one or more nodes fail.

III. OVERALL CONNECTION RELIABILITY

Let there exist a segmented backup path set between a pair of nodes and D. To make the analysis easier, assume that each segmented backup path protects only a single node of the primary path. Let us start with the smallest possible

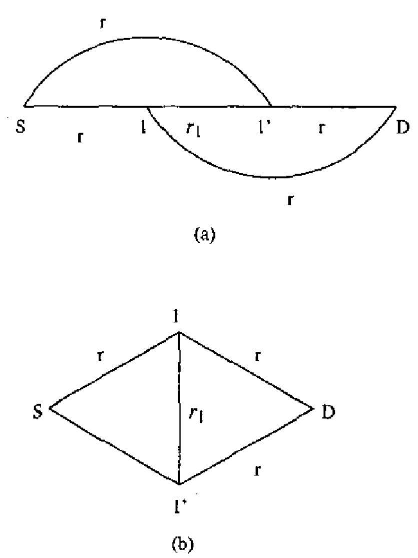

Fig. 7. A network of 4 nodes with two segmented backup paths.

set. This can be obtained by a network of four nodes and five edges as shown in Figure 7. For clarity, the Figure 7(a) can be redrawn as Figure 7(b). Assume, every link (except (1, 1') which has reliability r_1) has identical reliability r. Then every link has failure probability = 1 - reliability. Assume that the network will detect path failures and switch to an available alternative paths with probability 1. Let $F_{overall}$ and $R_{overall}$ represent the overall connection probability of failure and success from S to D, respectively. Obviously, $R_{overall}$ = 1 - $F_{overall}$.

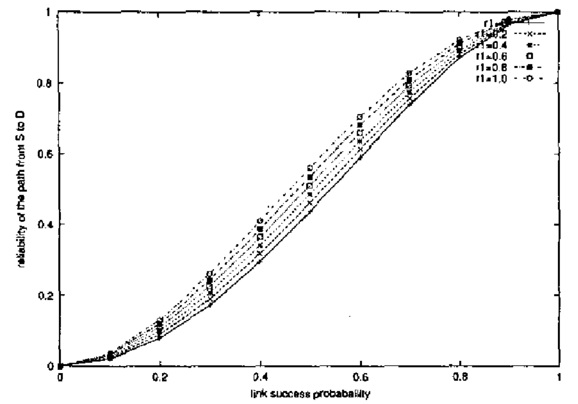

Fig. 8. link reliability vs. path reliability between S and D with varying values of r_1.

$R_{overall}(2)$ = Reliability of 2 disjoint paths $\langle S,1,D\rangle$ and $\langle S,1',D\rangle$ + Reliability of shared paths $\langle S,1,1',D\rangle$ and $\langle S,1',1,D\rangle$.
$= 1 - (1 - r^2)(1 - r^2) + 2r_1(1-r)^2r^2$
$= r^2(2-r^2) + 2r_1(1-r)^2r^2$.

Clearly the second term is due to the extra shared link (1, 1') and shows that the reliability of a network having two segmented backup paths is better than the reliability of a network having only two node disjoint paths. If $r_1 = 0$, then the overall reliabilty will be the reliability of two node disjoint paths.

Figure 8 shows the effect of r_1 on connection (path) reliability as link reliability increases in case of two segmented paths. Formula $R_{overall}(2)$ was used in calculating the results shown in Figure 8. It illustrates that the overall reliability increases as link reliability increases. It also illustrates that when the link reliability r=0.5, the overall reliability grows faster with the increase in r_1.

Now, consider a network which has three segmented backup paths, where first and last bridge protects a single node and an intermediate bridge protects two nodes on the primary path. $R_{overall}(3)$ = Reliability of 2 disjoint paths $\langle S,1,2,,D\rangle$ and $\langle S,1',2',D\rangle$ + Reliability of shared paths $\langle S,1,1',2',D\rangle$, $\langle S,1',1,2,D\rangle$, $\langle S,1,2,2',D\rangle$, $\langle S,1',2',2,D\rangle$, $\langle S,1,1',2',2,D\rangle$, and $\langle S,1',1,2,2',D\rangle$.
$= r^3(2-r^3) + 2(1-r)^3r^3(r_1 + r_2 - r_1r_2)$. Figure 10 shows

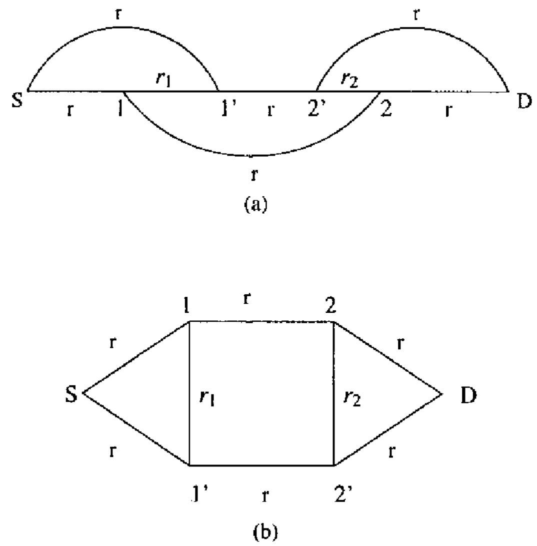

Fig. 9. A network of 6 nodes with 3 segmented backup paths.

the effect of r_1 on connection (path) reliability as link reliability increases in case of three segmented backup paths. Formula $R_{overall}(3)$ was used in for the results shown in Figure 10. It illustrates that the overall reliability increases as link reliability increases. Further, it also shows that when the link reliability r=0.5, the overall reliability grows but relatively slower as compared to $R_{overall}(2)$ with the increase in $r_1 (= r_2)$.

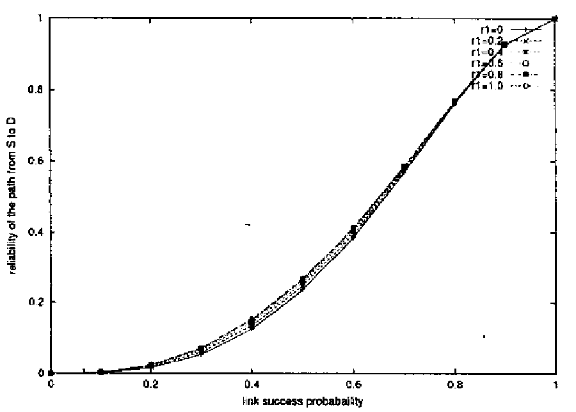

Fig. 10. link reliability vs. path reliability between S and D with varying values of r_1.

In general if a network has n segmented backup paths, where each intermediate bridge (not containing S and D) protects two nodes on the primary path. The first and last segmented backup path protect only one node each. Then, similary, connection reliability $R_{overall}(n)$ can given by

$$R_{overall}(n) = r^n(2 - r^n) + 2(1 - r)^n r^n (\sum_{1 \le i \le n-1} r_i - \sum_{1 \le i_1 < i_2 \le n-1} r_{i_1} r_{i_2} + \sum_{1 \le i_1 < i_2 < i_3 \le n-1} r_{i_1} r_{i_2} r_{i_3} - \ldots + (-1)^{n-2} r_{i_1} r_{i_2} \ldots r_{i_{n-1}}).$$

IV. CONCLUSIONS

In this paper, we have analysed the communication reliability of the segmented backup path *set* used by the SBSR ([6]) scheme. This scheme uses one of the backup paths in case a node along the primary path fails. The most important points that are made by this scheme are two. First, one is able to identify backup paths for any selected primary path, as long as there exists (any) pair of node disjoint paths from source to destination. In other words, if there are two node disjoint paths, one is not forced to use either of them to be the primary path. This has the advantage that the primary path may be selected based on QoS considerations rather than a consideration of fault tolerance. The second major advantage is that the communication reliability of segmented backup path set is higher as compared to two disjoint paths between a pair of nodes.

REFERENCES

[1] Nasipuri, A., Das, S.R.: On-demand multipath routing for mobile ad hoc networks. Proceedings of 8th Int. Conf. on Computer, Commun. and Networks, IEEE ICCCN'99 (1999) 64–70

[2] Karina, M.K., Das, S.R.: On-demand multipath distance vector routing in ad hoc networks. In Proceedings of IEEE International Conference on Network Protocols (ICNP), Riverside (2001) 14–23

[3] Nasipuri, A., Castaneda, R., Das, S.R.: Performance of multipath routing for on-demand protocols in mobile ad hoc networks. ACM/Baltzer MONET Journal **6** (2001) 339–349

[4] Lee, S.J., Gerla, M.: Aodv-br: Backup routing in ad hoc networks. Proceedings of IEEE WCNC (2000) 1311–1316

[5] Papadimitratos, P., Haas, Z., Sirer, E.: Path set selection in mobile ad hoc networks. Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (2002) 1–11

[6] Agarwal, A., Jain, B.: Qos-aware segmented backup source routing in mobile ad hoc networks. Proceedings of the second Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (2004) 122–131

[7] Gupta, A., Jain, B.N.: Qos-aware path protection schemes for mpls networks. Proceedings of the International Conference on Computer and Communication (ICCC) (2002) 103–118

[8] Banerjea, A.: Simulation study of the capacity effects of dispersity routing for fault-tolerant real-time channels. Proceedings ACM SIGCOMM (1996) 194–205

[9] Kao, B., Garcia-Molina, H., Barbara, D.: Aggressive transmissions of short messages over redundant paths. IEEE Transactions on Parallel and Distributed Systems **5** (1994) 102–109

[10] Zheng, Q., Shin, K.G.: Fault-tolerant real-time communication in distributed computing systems. IEEE Fault-Tolerant Computing Symp. Dig. Papers (1992) 86–93

[11] Grover, W.: The self-healing network: A fast distributed restoration technique for networks using digital crossconnect machines. Proceedings IEEE GLOBECOM (1987) 1090–1095