Analytical analysis of fault management in MANETs

Amadou Baba Bagayoko, Riadh Dhaou, Béatrice Paillassa

Analytical Analysis of Fault Management in MANETs

Amadou Baba BAGAYOKO[†],[*] Riadh DHAOU[†] and Beatrice PAILLASSA[†]
[†]University of Toulouse, IRIT Laboratory ENSEEIHT, 2, rue Camichel 31071 Toulouse, France
Email: {amadoubaba.bagayoko, riadh.dhaou, beatrice.paillassa}@enseeiht.fr
[*]Institut Telecom / Telecom Bretagne, 2 Rue de la Chataigneraie, CS 17607, 35576 Cesson Sevigne cedex, France

***Abstract*—This paper studies the fault management in mobile ad hoc networks. We focus on schemes which provide the robustness in MANETs: service restoration and service protection. Firstly, we propose an analytical comparison of service restoration, in terms of restoration time, at the routing layer (reactive and proactive protocols). Service restoration with reactive protocol is the most suitable in MANETs. The second motivation of the paper is an analytical study of the service protection at the network layer (multipath routing protocols) and its interest in terms of restoration time and reliability. The advantage of route redundancy in terms of restoration depends on the probability that alternate route is running after the primary route failure. The value, of this probability is studied, according to the used link connectivity maintenance mechanisms: Link-Layer Feedback Acknowledgements (LLF), Hello messages and Network-Layer Acknowledgements [1]. In terms of reliability, the multipath protocols are always better than unipath protocols. Segment recovery is the best recovery policy in terms of reliability.**

***Keywords*—mobile ad hoc network; robustness; protection and restoration services; analytical formulation: recovery policy; reliability.**

I. Introduction

Due to the unreliable characteristics of wireless communications, and nodes mobility, Mobile Ad hoc Networks (MANETs) suffer from frequent failures and reactivation of links. Consequently, the routes frequently change, causing significant number of routing packets to discover new routes, leading to increased network congestion and transmission latency. In order to reduce the frequent communication failures, robustness are introduced in MANETS by using either protection or restoration services. The two services differ mainly in their implementation timing: before or after the failure. Protection service is implemented before failure (usually during the initialization of the communication phase) in order to prevent, anticipate and reduce communication failures occurrence; while the restoration service is performed after the detection of communication failure, to quickly restore the service between endpoints.

In this paper, we adopt a robustness approach in order to improve communication performance. We analyze both services which provide the robustness in MANETs. We propose and study a protection architecture (by routes redundancy) which is coupled with a routing level restoration. The routing protocol is responsible of the failure detection phase, and uses various mechanisms either link-level notifications or network level notification to detect link failures.

Our first contribution is to propose an analytical comparison of the service restoration at the routing layer. We analyze the restoration time when either a reactive or a proactive unipath routing protocol is used. The best suitable category of routing protocol is derived from a comparison between the restoration time given at routing layer. Objective is to reduce the restoration time by choosing the adequate level in which reliability schemes should be activated.

The second contribution is based on multipath routing protocol. It is a protection mechanism based on route redundancy. In this architecture, the recovery operation is either to switch the traffic to an alternate route or to compute a new route. We propose an analytical comparison between different recovery policies of multipath routing protocol. We deduce that segment recovery is the best recovery policy in terms of recovery time and reliability. Analysis is based on an analytical formulation [2] that computes link reliability between adjacent nodes. This formulation takes into account nodes mobility model and the wireless communication characteristics including collisions between packets and signal attenuations. Nodes mobility model is Random Walk.

The rest of paper is organized as follows. Section II compares the restoration times with reactive and proactive routing protocols. Section III focuses on the interest of routes redundancy in MANETs according to restoration time and reliability. For each performance evaluation metric (restoration time and reliability), we propose an analytical formulation of unipath and two recovery policies of multipath (end-to-end and segment recovery).

II. Restoration on the routing layer

In mobile ad hoc networks, routing protocols are classified into three main categories: reactive, proactive and hybrid protocols. These categories differ depending on how the nodes obtain and maintain their routes. Proactive protocols actively maintain routes to reach all nodes in the network, while reactive protocols compute and maintain the route only when a data transmission is needed. Hybrid routing protocols try to combine the advantages of proactive and reactive routing: a proactive approach is adopted in the vicinity of the source and a reactive approach is used for the distant nodes of the source. Whatever routing protocol category is, link failure between two adjacent nodes on a route induced the failure of this route.

A. Reactive routing protocol

In reactive protocol, after the link failure detection, the upstream node of faulty link begins failure notification by sending

RERR (Route ERRor) packet to the recovery node (usually the source node). The RERR travels, hop-by-hop, from the upstream node of faulty link to the recovery node. Then, the detection time of the route failure $T_{d_{Route}}$ is composed of the detection time of communication failure between two adjacent nodes (n_i and n_{i+1}) and the time of notification phase T_n. Upon the reception of a RERR, source performs the recovery operation which consists on building a new route between two endpoints (T_{ro} is equal to T_{RD} route discovery time). After that, source makes traffic recovery by switching its traffic on the new route. The service restoration time for reactive routing protocol is:

$$\begin{aligned} T_{R_{Reac}} &= T_{d_{Route}} + T_{ro} + T_{tr} \\ &= T_{d_{(n_i,n_{i+1})}} + T_n + T_{RD} + T_{tr} \end{aligned} \quad (1)$$

B. Proactive routing protocol

In proactive protocol, the phases of fault detection and recovery operation are done simultaneously according to the topology information. When a node detects a communication failure with one neighbor, it modifies its routing table and waits for the next broadcast period to communicate its new topology information to its other neighbors. Topology information is periodically broadcasted hop-by-hop. Source node, like all the other nodes, updates its routing table.

The source switches the traffic on a new non-faulty route, if it exists. Restoration time, as illustrated in Figure 1, is composed of the detection time of communication failure between two adjacent nodes (n_i and n_{i+1}), the time T_{Bcast} (sum of the remaining times of broadcasts periods T_{remain_j}) and the time of notification phase T_n (sum of the propagation time of topology information between the broadcasting nodes):

$$T_{R_{Proac}} = T_{d_{(n_i,n_{i+1})}} + T_{Bcast} + T_n + T_{tr} \quad (2)$$

Assume that there are M intermediate nodes between the node detecting the failure n_j and the source which broadcasts topology information, the time T_{Bcast} can be expressed as follows:

$$T_{Bcast} = \sum_{j=0}^{M} T_{remain_j} \quad (3)$$

where T_{remain_j} is the remaining time of broadcasts periods of node j.

$$T_n = \left[\sum_{j=0}^{M} T_{Propagation_{(j,j+1)}} \right] \quad (4)$$

Service restoration in hybrid routing protocol depends on the location of the faulty link which causes the communication failure. When the faulty link is in the vicinity of the data source, hybrid routing protocol will act as a proactive routing protocol, otherwise it adopts the reactive protocol behavior in the restoration.

C. Restoration times comparison at routing layer

In this part, we compare the restoration times of reactive and proactive routing protocol. The duration of the service restoration of a routing protocol depends on the duration

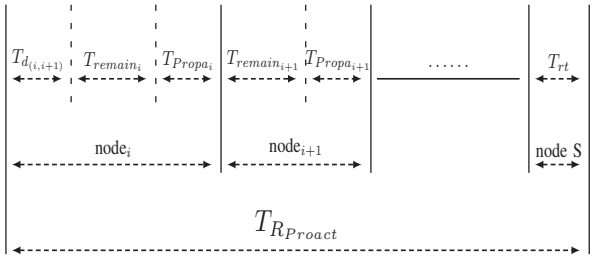


Fig. 1. Service restoration of proactive protocol

phases of the link failure detection, fault notification and recovery operation because the traffic recovery phase is almost instantaneous ($T_{tr} \approx 0$).

The comparison of the recovery time between reactive and proactive protocols is a comparison between the route discovery phase delay T_{RD} and the time T_{Bcast} (sum of the remaining times of broadcasts periods), according to the equations 1 and 2:

$$\begin{aligned} T_{R_{Proac}} - T_{R_{Reac}} &= T_{Bcast} - T_{RD} \\ &= \left[\sum_{j=0}^{M} T_{remain_j} \right] - T_{RD} \end{aligned} \quad (5)$$

Reactive routing protocols like AODV [3] and DYMO [4] define the maximum of route discovery time to 2 seconds ($T_{RD} \leq 2$ sec). While the value of T_{Bcast} is not predictable and is generally higher than the route discovery time of reactive protocols. Its value depends on the number of intermediate nodes between the node of the detection of link failure and the source node, and their state at the receipt time of the failure notification.

The first approach for a quick restoration in proactive protocols, consists on reducing the length time of the broadcast period of the topology information. A second solution is to sequentialize the phases of service restoration *i.e* after the link failure detection, upstream node of faulty link begins immediately the notification phase without waiting the end of its period. The common drawback of both approaches is a significant increase in the number of control messages.

Restoration time with reactive protocols is bounded and usually less than those of proactive protocols. In next, we focus on the reactive routing protocols (unipath and multipath).

III. Analytical study of route redundancy

The main limitation of the standard reactive routing protocols is that they build only one route between a source and a destination. These are called *unipath* protocols. After the occurrence of a communication failure on an active route, a restoration must be performed to find a new route between endpoints. Data packets sent by the source node, before it is informed of the fault state (detection and notification phase), will not reach their destination. Intermediate nodes (which receive route failure notification prior to source) drop data packets to unreachable destination because they do not have an

alternative path. Upon failure notification receipt, the source stops to send data packets to this destination. Application traffic is recorded on routing queue depending on the traffic amount and the queue size. This may cause also data packets losses and congestion. If the source node communicates again with the same destination, it initiates a new route discovery. The route discovery increases the end-to-end delay of packets and the number of control message.

In order to increase the packet delivery ratio (PDR) and decrease service restoration time, multipath routing protocols based on unipath standard protocols have been proposed: AODV-BR [5], SMR [6], AOMDV [7], [8], DYMOM [9] and MDYMO [10]. Multi-path protocols use the notion of redundancy, they establish and maintain one primary route and one (or more) alternate route between the source and destination. Traffic is switched immediately on the alternate route (without new route discovery) when primary route fails.

These proposals show by simulation the advantage of multipath approach in terms of packet delivery ratio (PDR), of control messages (overhead), and end-to-end delay. In addition to these works, we propose an analytical comparison of unipath and multipath approaches in terms of restoration time and reliability. We derive the conditions under which the multipath approach improves the network performance.

Depending upon the disjointness of primary and alternate route, different recovery policies can be envisaged: the link-to-link recovery, the segment recovery and the end-to-end recovery. For the first one, the traffic is sent to the alternate link while for the two others the traffic is moved not only on a secondary link but also on another route. When the new route is entirely disjoint from the previous primary route, the recovery is an end-to-end recovery. On the contrary, in case of common elements between the two routes, we refer to a segment recovery. Obviously, the adoption of a given recovery policy depends on the network topology. For low density networks, where the probability to obtain a large number of disjoint routes is small, the robustness obtained from an end-to-end recovery would not be so interesting. However, considering a network topology where all the recovery policies are applicable, one redundancy policy would more improve the network reliability than an other one. Therefore, the problem is to determine the right level of protection for the ad hoc network.

In [11], we have studied and compared three recovery policies in heterogeneous networks in which each node has Zigbee and WiFi technologies. In this paper, we are interested in homogeneous MANET (*i.e*, all nodes have the same technology and each node has only one technology). Possible recovery policies in homogeneous MANET are end-to-end and segment recovery.

The objective of this section is to answer the following questions:

- What provides the multipath routing protocols and when are they interesting ?
- What are the advantages and disadvantages of the two recovery policies ?
- What type of recovery policies is suitable according to the mechanism of links failure detection ?

In first, we propose an analytical formulation of service restoration time of two recovery policies of multipath routing protocols. We then show the contribution of multipath at routing layer in terms of recovery time.

In the second part, we study the two policies of multipath routing protocols in terms of reliability. Analytical formulation of these recovery policies in terms of reliability are proposed and compared to unipath recovery.

A. Service Restoration Time: Reactive Unipath and Multipath

We focus on multipath routing protocol approach in which the primary route is used for the traffic transfer until a failure occurrence; and the traffic will be switched on an alternate route after the failure detection of the primary route. In this approach, route redundancy is used in order to reduce the restoration time then its interest depends on the probability that the alternate route is still functioning, *i.e.*. active, when the primary route is failed. To study the gain of route redundancy, we compare restoration time of multipath and unipath routing protocols.

1) Unipath reactive routing protocol: We continue the formulation of restoration time of unipath reactive routing protocol based on equation 1:

$$T_{R_{Reac}} = T_d + T_{n[n_i \rightarrow S]} + T_{RD} + T_{tr} \tag{6}$$

where n_i is the node which has detected communication failure; T_d is the time to detect link failure between any two adjacent nodes; $T_{n[n_i \rightarrow S]}$ is the notification time of link failure detected by n_i to source S.

2) Multipath reactive routing protocol: In this analysis, we assume that recovery node has two routes (one primary Ro_{pri} and one alternate Ro_{sec}) which are node-disjoint or link-disjoint according to recovery policy.

Restoration service of multipath protocols is slightly different from that of *unipath* protocols. It depends on the state (failed or not) of the alternate route Ro_{sec} when recovery node receives the failure notification of the primary route Ro_{pri}. Let $p_{secVald}$, the probability that the alternate route is active after the recovery node N_{rec} receives the failure notification RERR of the primary route, the restoration time with multipath is:

$$\begin{aligned} T_{R_{MP}} = \left(T_d + T_{n[n_i \rightarrow S]}\right)_{Pri} + [p_{secVald} \times \{T_{R_{MP}}\}_{Case1}] \\ + [(1 - p_{secVald}) \times \{T_{R_{MP}}\}_{Case2}] \end{aligned} \tag{7}$$

Case 1: The recovery node immediately switches data traffic on the alternate route Ro_{sec} which is functioning after the receipt of *RERR*

$$\{T_{R_{MP}}\}_{Case1} = T_{tr} \tag{8}$$

Case 2: Alternate route Ro_{sec} is also broken but recovery node does not know this information. As in case 1, it switches the data traffic on alternate route which becomes active. Data transmission on an alternate route will lead to the detection of its failure by the adjacent nodes to faulty component (link or node). When recovery node receives failure notification *RERR* of its alternate path, it initiates a new route discovery, and does traffic recovery.

$$\{T_{R_{MP}}\}_{Case2} = \left(T_{tr} + T_d + T_{n[n_j \rightarrow S]}\right)_{Sec} + T_{RD} + T_{tr} \tag{9}$$

where node n_j detects alternate route failure and initiates notification phase.

a) End-to-end recovery policy: In this policy, the data source S is the recovery node, then its restoration time is:

$$\begin{aligned} T_{R_{MP(E2E)}} &= \left(T_d + T_{n[n_i \to S]}\right)_{Pri} + (p_{secVald} \times T_{tr}) \\ &+ (1 - p_{secVald}) \times \left[\left(T_{tr} + T_d + T_{n[n_j \to S]}\right)_{Sec} + T_{RD} + T_{tr}\right] \\ &= \left[(1 - p_{secVald}) \times T_{R_{UP-Reac}}\right] + \left(T_{tr} + T_d + T_{n[n_i \to S]}\right) \end{aligned} \tag{10}$$

where $T_{n[n_i \to S]} = T_{n[n_j \to S]}$ (assuming for simplification reason that, hops number $n_i \to S$ and $n_j \to S$ are equal).

b) Segment recovery policy: Intermediate node NI (between node of failure detection n_i and source S) switches data traffic on an alternate route.

$$\begin{aligned} T_{R_{MP(SR)}} &= \left[\left(T_d + T_{n[n_i \to NI]}\right)_{Pri} + T_{RD} + T_{tr}\right] \\ &- (p_{secVald} \times T_{RD}) + \left[(1 - p_{secVald}) \times \left(T_{tr} + T_{d_{Route}}\right)_{Sec}\right] \end{aligned} \tag{11}$$

Node NI is nearer to the node detecting failure n_i than source S:

$$\begin{aligned} T_{n[n_i \to NI]} &< T_{n[n_i \to S]} \\ T_d + T_{n[n_i \to NI]} &< T_d + T_{n[n_i \to S]} \\ T_{R_{MP(SR)}} &< T_{R_{MP(E2E)}} \end{aligned} \tag{12}$$

Therefore, restoration time of the segment recovery $T_{R_{MP(SR)}}$ is less than the one of end-to-end recovery $T_{R_{MP(E2ER)}}$.

3) Contribution of route redundancy: In order to assess the advantage of the routes redundancy in terms of adaptation to the topology changes, we compare restoration time with unipath protocol (equation 6) and end-to-end recovery policy of multipath protocol (equation 10). Restoration time with unipath protocol is higher, if:

$$\begin{aligned} T_{R_{Unipath}} - T_{R_{MP}} > 0 \quad &\Rightarrow \\ p_{secVald} > \frac{T_{tr} + T_d + T_n}{T_d + T_n + T_{RD} + T_{tr}} \end{aligned} \tag{13}$$

Traffic recovery is almost instantaneous ($T_{tr} \approx 0$). Lets assume, n_i the node which detects the communication failure. This node n_i initiates the faults notification phase by sending RERR to the source node S. Assuming that the failure occurs halfway between the source and destination, the notification phase time (between nodes *i* and S) can be approximated by the half of the travel time of the control messages (RREQ or RREP) between the source S and the destination D (or inversely).

$$T_{n[n_i \to S]} = T_{RERR[n_i \to S]} \approx \frac{T_{RREQ[S \to D]}}{2} \approx \frac{T_{RREP[D \to S]}}{2} \tag{14}$$

Therefore, the route discovery time T_{RD} is:

$$\begin{aligned} T_{RD} &= T_{RREQ[S \to D]} + T_{RREP[D \to S]} \\ &\approx 4 \times T_{RERR[n_i \to S]} \end{aligned} \tag{15}$$

Expression 13 can rewritten by using equations 14 and 15, as follows:

$$\begin{aligned} p_{secVald} &> \frac{T_d + \frac{T_{RD}}{4}}{T_d + \frac{T_{RD}}{4} + T_{RD}} \\ &\approx \frac{(4 \times T_d) + T_{RD}}{(4 \times T_d) + (5 \times T_{RD})} \end{aligned} \tag{16}$$

The value of the probability $p_{secVald}$ depends on T_d which is computed according to the used link connectivity maintenance mechanisms (Link-Layer Feedback Acknowledgements(LLF), Hello messages, Network-Layer Acknowledgements (NA) [1]).

a) Link-Layer Feedback (LLF): Routing protocol uses link layer notification at the MAC layer (like in IEEE 802.11) to detect link failure. In IEEE 802.11 model, the MAC protocol sends CTS (Clear-To-Send) in response to RTS (Request-To-Send) and ACK in response to a received data packet. In case of non reception of the acknowledgment (respectively CTS frame) after a timeout, the MAC protocol of the data (respectively of RTS) sender enters in *hold-off* period and forwards the data packet (respectively RTS). The purpose of the hold-off period is to be sure of the fault state. If no acknowledgment is received after the maximum number of retransmission attempts (default value of retryLimit is 7), the MAC protocol detects a failure (MAC address unreachable). An indication mechanism between layers 2 and 3 enables to the routing protocol to detect the link failure.

Failure detection time using the Link-Layer Feedback (LLF) is composed of the waiting time T_W on node n_i of a packet to be transmitted to the node n_{i+1} and the time of the maximum number of retransmission attempts $T_{retrans}$:

$$E\{T_d\}_{(LLF)} = T_W + T_{retrans} \tag{17}$$

Waiting time T_W may be viewed as the travel time of data packet between source and node of link failure detection. It is approximately equal to the notification phase time (transmission of an RERR message from the failure detection node to the source).

$$T_W \approx T_{RERR} \tag{18}$$

The duration of the maximum number of retransmission attempts $T_{retrans}$ is very small (in the order of millisecond) compared to T_W. Therefore, detection time using the Link-Layer Feedback (LLF) is:

$$E\{T_d\}_{(LLF)} \approx T_W \approx T_{RERR} \approx \frac{T_{RD}}{4} \tag{19}$$

Using equations 15 and 19 in expression 16, we obtain:

$$\{p_{secVald}\}_{(LLF)} > \frac{\left(4 \times \frac{T_{RD}}{4}\right) + T_{RD}}{\left(4 \times \frac{T_{RD}}{4}\right) + (5 \times T_{RD})} \approx \frac{1}{3} \tag{20}$$

The restoration of the multipath routing protocol is faster than that of unipath protocol, using Link-Layer Feedback Acknowledgement to detect link failure, if $p_{secVald} > \frac{1}{3}$.

b) Hello Messages: This well-known mechanism allows nodes to maintain their neighbor knowledge table. Each node broadcasts Hello message every interval HELLO_INTERVAL(HI). Each node has a neighbors table which is regularly updated. A node removes one neighbor from its table if it does not receive any Hello message or data packets from this neighbor during *AHL* $\times$ *HI*; ALLOWED_HELLO_LOSS (AHL) is the maximum number of attempts before failure detection.

Since the failure instant is a priori unknown, authors of [12] characterize $E\{T_d\}_{(Hello)}$ as a uniformly distributed random variable between HI and AHL $\times$ *HI*, as follows:

$$E\{T_d\}_{(Hello)} = \frac{1 + AHL}{2} \times HI = 1.5sec \qquad (21)$$

Using experimental default values from RFC 3561 [3]: $AHL = 2$ and $HI = 1000$ milliseconds.

Equation 15 is rewritten, for Hello Messages, by using equation 21:

$$\{p_{secVald}\}_{(Hello)} \quad > \quad \frac{6 + T_{RD}}{6 + (5 \times T_{RD})} \qquad (22)$$

c) Network-Layer Acknowledgements (NA)[1]: Dynamic Source Routing DSR Protocol [1] proposes this link connectivity maintenance mechanism in the absence of other available acknowledgement mechanism. Each forwarding node of data packet must verify the reachability of the next-hop node on the active route. To do so, node inserts an Acknowledgement Request option in the DSR Options header of the packet. For example, forwarding node n_i set this option before send of the data packet to its next-hop node n_{i+1}. When the node n_i receives ACK from the node n_{i+1}, it may choose to not requiring ACKs from the same neighbor for a period of time equal to *MaintHoldOffTime* (default value is set to 250 ms). The maximum number of ACK request attempt is *MaxMaintRexmt* (equal to 2 by default). Sender detects the link failure to the next hop, if the maximum number of ACK requests have been transmitted and no ACK has been received. The mean value of the failure detection time of this mechanism is:

$$E\{T_d\}_{(NA)} = \frac{MaintHoldOffTime}{MaxMaintRexmt} = 0.125 \text{ sec} \qquad (23)$$

Therefore, we obtain the below expression, by using the previous value in equation 15:

$$\{p_{secVald}\}_{(NA)} \quad \geq \quad \frac{0.5 + T_{RD}}{0.5 + 5 \times T_{RD}} \qquad (24)$$

d) Advantage of route redundancy according to link connectivity mechanism: Contrarily to intuition, the route redundancy is not always better than unipath in terms of restoration time.

In case of link-layer notification, the route redundancy is better than unipath if the probability $p_{secVald}$ is higher than $1/3$. Note that, in link notification case the probability $p_{secVald}$ is not depending on the route discovery time T_{RD} contrarily to the two other mechanisms.

Concerning these mechanisms, figure 2 shows the impact of route discovery time T_{RD} on the threshold of probability

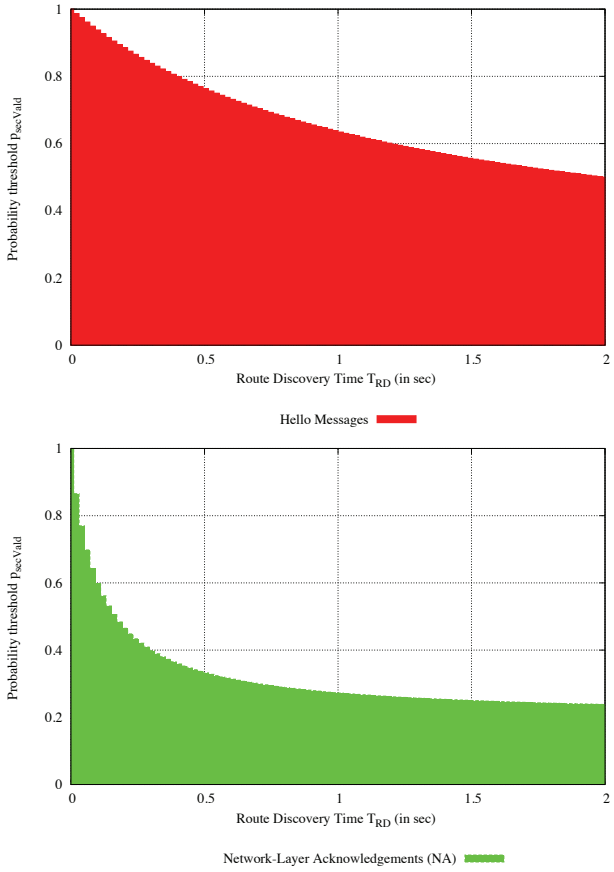


Fig. 2. Probability threshold $p_{secVald}$ vs Route Discovery Time T_{RD}

$p_{secVald}$ (equations 22 and 24). We observe that the increase of the route discovery time T_{RD} decreases the threshold value of the probability $p_{secVald}$. The advantage of route redundancy is correlated with T_{RD} and it is not interesting for short route discovery time ($T_{RD} < 0.5$).

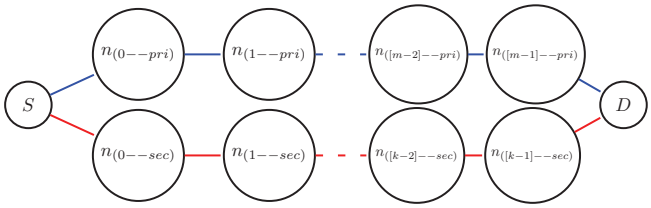
B. Reliability of recovery policies

In this section, we propose an analytical formulation of route reliability of unipath and multipath (end-to-end and segment recovery) routing protocols.

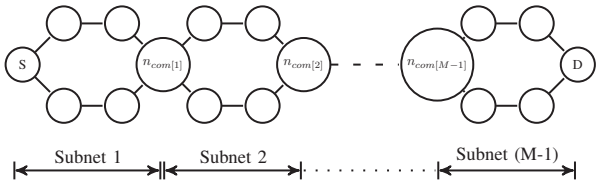
1) Analytical formulation: We use the Reliability Block Diagram (RBD) in order to formulate reliability of recovery policies. Set of sequential elements reliability is the product of the element reliability (on the contrary, when elements are in parallel order, the product concerns the unreliability). Notations:

- R_{n_i}: reliability of node n_i
- R_{L_i}: reliability of link L_i between nodes n_i and n_{i+1}

a) Unipath route reliability: Route reliability between a source n_0 and a destination n_m composed of m links and $(m + 1)$ nodes is equal product of reliability of all links and



(a) End-to-end recovery



(b) Segment recovery

Fig. 3. Recovery policies

nodes on this route [13]:

$$R_{Ro_{(n_0 \leftrightarrow n_m)}} = R\left[L_0 \cap L_1 \cdots \cap L_{m-1}\right] \times \prod_{i=0}^{m} R_{n_i} = R_{n_m} \times \prod_{i=0}^{m-1} \left[R_{n_i} \times R_{L_i}\right] \quad (25)$$

b) End-to-End recovery: Reliability of end-to-end recovery named $R_{E2E_{(S \leftrightarrow D)}}$ is function of the reliability of primary and alternate routes between source S and destination D (figure 3(a)). To simplify the above equations, we omit to precise that the reliability is between the source S and the destination D; for example $R_{E2E_{(S \leftrightarrow D)}}$ is noted R_{E2E} . There are two common nodes between two routes: the end-points S and D. We obtain the following expression by using the decomposition in series/parallel of Block Diagram:

$$R_{E2E} = R_S \times R_D \times \left[1 - \left(1 - \tilde{R}_{Ro_{pri_{(S \leftrightarrow D)}}}\right) \times \left(1 - \tilde{R}_{Ro_{sec_{(S \leftrightarrow D)}}}\right)\right] \quad (26)$$

where:

- Reliability $\tilde{R}_{Ro_{pri}}$ is reliability product of all components on the **primary route** except that of the source S and the destination D. We assume that primary route has m links, and note nodes on this route as follows $n_{(j--pri)}, \forall\ (0 \leq j \leq m-1)$.

$$\tilde{R}_{Ro_{pri}} = R_{L_{\left[S, n_{(0--pri)}\right]}} \times R_{L_{\left[n_{([m-1]--pri)}, D\right]}} \times \left[\prod_{j=0}^{m-1} R_{n_{(j--pri)}} \times R_{L_{(j--pri)}}\right] \quad (27)$$

- Reliability $\tilde{R}_{Ro_{sec}}$ is reliability product of all components on the **alternate route** except of source S and destination D. We assume that the alternate route has k links, and note nodes on this route as follows $n_{(j--sec)}, \forall\ (0 \leq j \leq k-1)$.

$$\tilde{R}_{Ro_{sec}} = R_{L_{\left[S, n_{(0--sec)}\right]}} \times R_{L_{\left[n_{([k-1]--sec)}, D\right]}} \times \left[\prod_{j=0}^{k-1} R_{n_{(j--sec)}} \times R_{L_{(j--sec)}}\right] \quad (28)$$

End-to-End recovery policy resists to simultaneous failures of one or more components (links, nodes) on the primary path. The recovery process may be implemented in various routing protocols (source routing and hop-by-hop). Although the complexity of the recovery is reduced since it is supported by the end-point nodes, this policy suffers from some drawbacks. As the recovery domain is the full path, it increases the total time of the recovery process. Moreover, in case of a simultaneous link failure on the primary and the secondary paths, the recovery is not possible.

c) Segment recovery: Segment recovery policy can be used if the primary and the the alternate have at least an other one common node different from the source and destination; routes are link-disjoint (figure 3(b)). In order to formulate the reliability of this policy, we divide into $(M-1)$ subnets; where M is the number of common nodes between the primary and the alternate routes. Subnet i starts from the common node $n_{com[i-1]}$ and ends to the next common node $n_{com[i]}$ between two routes (primary and alternate). (Note: S and D are also respectively noted $n_{com[0]}$ and $n_{com[M]}$).

Reliability of each subnet is computed according to equation 26. Reliability of segment recovery is obtained by the multiplication of the reliability of $(M-1)$ sub-networks divided by the reliability of common nodes $n_{com[i-1]}$ $(\forall i, 1 \leq i \leq (M-1))$. The goal of this division is to avoid taking into account the reliability of one common node two times: the first time with subnet that it begins and the second time with subnet that it ends. Segment recovery reliability, composed of $(M-1)$ subnet is:

$$R_{SR} = R_{SR_{\left(n_{com[0]} \leftrightarrow n_{com[M]}\right)}} = \frac{\prod_{i=0}^{M-1} R_{E2E_{\left(n_{com[i]} \leftrightarrow n_{com[i+1]}\right)}}}{\prod_{i=1}^{M-1} R_{n_{com[i]}}} \quad (29)$$

where $R_{E2E_{\left(n_{com[i]} \leftrightarrow n_{com[i+1]}\right)}}$ is given from the equation 26.

Note that, when $M = 2$ (*i.e* common nodes between the primary and the alternate routes are only the source S and the destination D), equations 26 and 29 are equivalent.

Since, each recovery domain of the segment recovery is smaller than the one of the whole network (for a network with at least 2 segments), it is more rapid than the end-to-end recovery. Moreover, segment recovery may provide better

protection than the end-to-end one. When two simultaneous link failures affect both the primary and the secondary paths, the end-to-end recovery is not able to compute any path. In the same case, the segment recovery establishes the alternate path by using fault-free segments.

2) Performances evaluation: Equations 25, 26 and 29 depend on the reliability of two elements: link and node. These reliabilities are time-depend *i.e* they are computed during an interval (e.g $[t, t+T]$).

In [2], we have proposed an analytical model of link reliability where nodes move according to Random Walk Mobility model. Link reliability between two mobile nodes is computed in the time interval $I = [t, t+T]$ whose length T is equal to the communication duration. Communication time T is partitioned into k small intervals of fixed length time, termed *epoch*.

A Markov chain model is used to describe the distance evolution between nodes. This Markov chain is composed of $(n+1)$ states divided into two subsets: E_{S1} and E_{S2}. The first subset E_{S1} contains all possible states between two nodes whose distance is less than the theoretical transmission range r_0. The transmission range r_0 is divided into n equivalent length bins of width ε meters: $r_0 = n \times \epsilon$. Hence, the first subset E_{S1} is composed of the (n) first states of the Markov chain: $E_{S1} = \{e_1, e_2, \ldots, e_i, \ldots, e_n\}$. The second subset contains only the absorbing state $(n+1)$ which models the case where the distance between nodes is greater than r_0.

Link reliability $R_{L_{(a,b)}}(k)$ is defined as the product of (1) no packet drop probability $\bar{p}_{Coll}$ due successive collisions, and (2) the sum of probabilities that two nodes communicate without channel error from the state e_1 to the state e_n at the epoch k. In a given state e_i $(0 \leq i \leq n)$, the probability that two nodes communicate without channel error after k epochs is the product of (2.a) the probability $rd_i(k)$ that the distance between the pair of nodes after k epoch is equal to i *meters*, and (2.b) the probability $\bar{p}_{Channel}(i)$ that the packet sent on the distance of i *meters*, is not lost due to channel errors after m successive retransmissions.

Link reliability is formulated as follows:

$$R_L(k) = \sum_{i=1}^{n} cr_{iL}\,(k) \tag{30}$$

where $cr_{iL}\,(k)$ are the elements of the reliability distance probability vector $CR_L(k)$. Each $cr_{iL}\,(k)$ represents the communication reliability between two nodes at distance i:

$$cr_{iL}\,(k) = \bar{p}_{Coll} \times rd_i(k) \times \left(\bar{p}_{Channel}\,(i)\right)^k \tag{31}$$

- p_{Coll}: probability that packet is dropped due to m successive collisions; $\bar{p}_{Coll} = 1 - p_{Coll}$
- $rd_i(k)$: probability that inter-node distance is equal to i meters after k epoch
- $p_{Channel}\,(i)$: probability that packet is dropped due to signal attenuation after m retransmissions between nodes separated by distance i; $\bar{p}_{Channel}(i) = 1 - p_{Channel}(i)$

Concerning the nodes reliability, we suppose that it varies slightly in the time intervals that interests us (< 100 sec). Thus, in this rest of this paper, we assume that the reliability of all nodes is constant and is equal to 1.0 ($\forall\ j\ R_{n_j} = 1$).

In the evaluation, we assume that during each route discovery the following assumptions are true:

- Primary and alternate routes have the same hop-count
- In segment recovery, there is exactly one common node between source S and destination D (*i.e* $M = 3$ thus there are 2 subnets). The common node is in halfway between S and D in term of hop-count.

Figure 4 compares the reliability of unipath and the two recovery schemas of multipath in function of the communication duration for values of hop-count N in the multipath fading environment where nodes according to speed $v_{max} = 10$ m/s and theoretical transmission range of each node $r_0 = 250$ m.

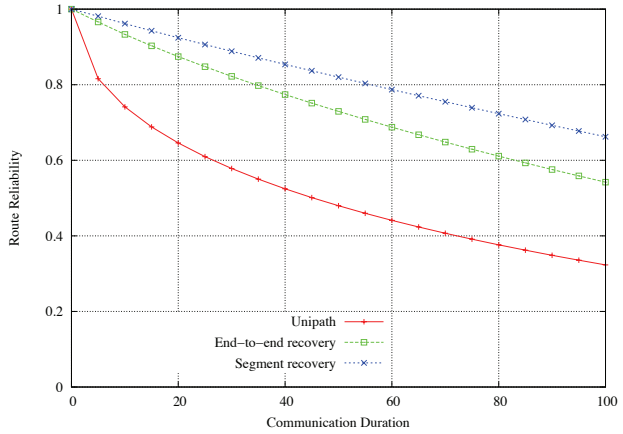
Expected results are observables. In all figures, for the three recovery schemas, the route reliability decreases with the communication duration growing. When N increases, the route reliability decreases because if there are more links between source and destination, probability of route breakage is more important too.

Concerning the recovery policies, we observe that the segment recovery is always the better recovery schema with different values of hop-count. Due to the size of the recovery domain that is smaller compared to the end-to-end recovery domain. End-to-end recovery is also always better than *unipath* routing protocol in terms of reliability.

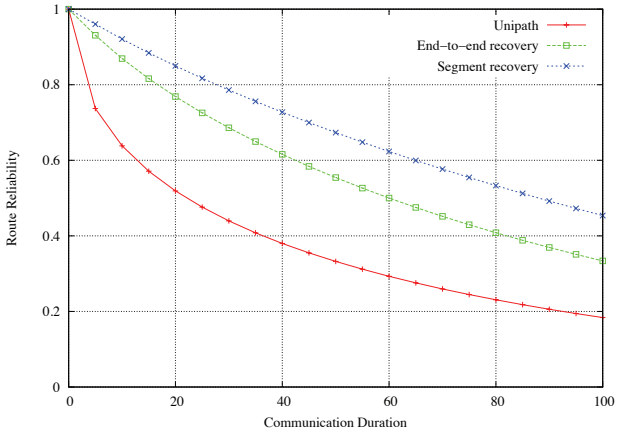
IV. Conclusion

The paper focuses on the robustness in ad hoc networks. We have compared unipath and multipath routing protocols in terms of restoration time and reliability. In terms of restoration time, the advantage of route redundancy depends on the probability $p_{secVald}$ that alternate route runs after the primary route failure. This probability depends on link connectivity maintenance mechanism. We analyze the interest of route redundancy of link failure detection mechanisms (Link-Layer Feedback (LLF), Hello messages, Network-Layer Acknowledgements). We show that route redundancy is not always better than unipath routing protocol in terms of restoration time. In case of link-layer notification, the route redundancy is better than unipath if the probability $p_{secVald}$ is higher than $1/3$. Concerning two other mechanisms, the advantage of route redundancy is correlated with T_{RD} and it is not interesting for short route discovery time ($T_{RD} < 0.5$). This is the case for small networks for localized traffic i.e between the next neighbours. It may be interesting to have adaptive approach in route discovery phase. For example, routing protocol may build only one route (unipath approach) when $T_{RD} < 0.5$ sec; else use route redundancy (multipath approach).

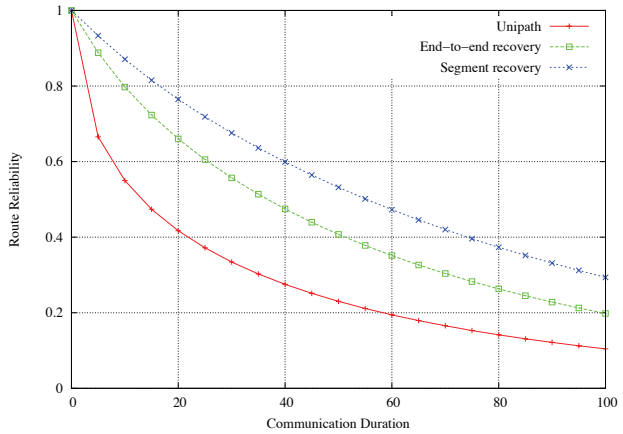
In terms of reliability, multipath routing is always better than unipath routing. Moreover, we show that segment recovery is the best recovery policy compare to the end-to-end recovery.



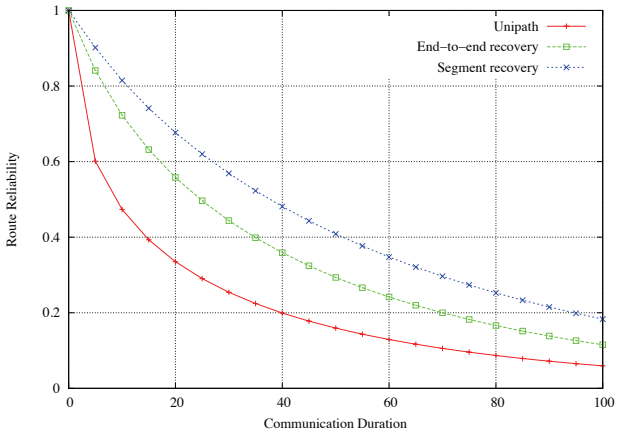
(a) Number of hops= 4



(a) Number of hops= 6



(b) Number of hops= 8



(b) Number of hops= 10

Fig. 4. Routes reliability of *unipath* and *multipath* (end-to-end and segment recovery): $v_{max} = 10$ m/s

REFERENCES

[1] D. Johnson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," RFC 4728 (Experimental), Internet Engineering Task Force, Feb. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4728.txt

[2] A. B. Bagayoko, "Robustness Policies in Mobile Ad hoc Networks," *Institut National Polytechnique de Toulouse*, vol. PhD, July 2012.

[3] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561 (Experimental), Internet Engineering Task Force, Jul. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[4] I. D. Chakeres and C. E. Perkins, "Dynamic manet on-demand (dymo) routing," Published Online, Jul. 2010, expiration: Janvier 2011. [Online]. Available: http://tools.ietf.org/id/draft-ietf-manet-dymo-21.txt

[5] S. J. Lee and M. Gerla, "AODV-BR: backup routing in ad hoc networks," *Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE*, vol. 3, pp. 1311–1316, 2000. [Online]. Available: http://dx.doi.org/10.1109/WCNC.2000.904822

[6] ——, "Split multipath routing with maximally disjoint paths in ad hoc networks," *Communications, 2001. ICC 2001. IEEE International Conference on*, vol. 10, pp. 3201–3205 vol.10, Aug. 2002. [Online]. Available: http://dx.doi.org/10.1109/ICC.2001.937262

[7] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," *Network Protocols, 2001. Ninth International Conference on*, pp. 14–23, 2001.

[8] ——, "Ad hoc on-demand multipath distance vector routing," *WCMC 2006, Wireless Communications and Mobile Computing*, vol. 6, no. 7, pp. 969–988, 2006. [Online]. Available: http://dx.doi.org/10.1002/wcm.432

[9] G. Koltsidas, F.-N. Pavlidou, K. Kuladinithi, A. Timm-Giel, and C. Gorg, "Investigating the performance of a multipath dymo protocol for ad-hoc networks," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, Sep. 2007, pp. 1–5.

[10] M. Nacher, C. Calafate, and P. Manzoni, "Multipath extensions to the dymo routing protocol," in *Mobile Wireless Communications Networks, 2007 9th IFIP International Conference on*, Sep. 2007, pp. 1 –5.

[11] A. B. Bagayoko and B. Paillassa, "Analysis of Robustness in Heterogeneous Ad Hoc Networks (regular paper)," in *International Wireless Communications and Mobile Computing Conference (IWCMC), Istanbul,Turkey, 05/07/2011-08/07/2011*, juillet 2011.

[12] C. Gomez, D. Mediavilla, P. Salvatella, X. Mantecon, and J. Paradells, "A study of local connectivity maintenance strategies of manet reactive routing protocol implementations," in *Wireless Communication Systems, 2006. ISWCS '06. 3rd International Symposium on*, Sep. 2006, pp. 228 –232.

[13] A. B. McDonald and T. Znati, "A mobility based framework for adaptive clustering in wireless ad-hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1466–1487, 1999.