ROMER: Resilient Opportunistic Mesh Routing for Wireless Mesh Networks

Yuan Yuan[1], Hao Yang[2], Starsky H.Y. Wong[2], Songwu Lu[2], William Arbaugh[1]
[1]Computer Science Department
University of Maryland
College Park, MD 97124
Email: {yuanyuan, waa}@cs.umd.edu
[2]Computer Science Department
University of California, Los Angeles
Los Angeles, CA 90095
{hyang, hywong1, slu}@cs.ucla.edu

***Abstract*— Wireless mesh networks hold promises to provide robust and high-throughput data delivery to wireless users. In a mesh network, high-speed Access Points (HAPs), equipped with advanced antennas, communicate with each other over wireless channels and form an indoor/outdoor broadband backhaul. This backbone efficiently forwards user traffic to a few gateway APs (GAPs), which additionally have high-speed connections to the wired Internet. In this paper, we describe ROMER, a resilient and opportunistic routing solution for mesh networks. ROMER balances between long-term route stability and short-term opportunistic performance. It builds a runtime, forwarding mesh on a per-packet basis that offers a set of candidate routes. The actual forwarding path by each packet opportunistically adapts to the dynamic channel condition and exploits the highest-rate wireless channels at the time. To improve resilience against lossy links, HAP failures or HAPs under DoS attacks, ROMER delivers redundant data copies in a controlled and randomized manner over the candidate forwarding mesh. We evaluate the effectiveness of ROMER through both simulations and analysis.**

I. INTRODUCTION

Wireless mesh networks seek to build a resilient and high-performance infrastructure to provide users pervasive Internet access. In a mesh network, each client accesses a local high-speed access point (HAP), and multiple stationary HAPs communicate with one another over the wireless channel and form a multihop, wireless backbone for data delivery. This backbone eventually forwards user traffic to a few gateway APs (GAPs) that additionally connect to the wired Internet. Compared with the current-generation single-hop wireless cellular infrastructure, mesh networks build a wireless broadband backhaul to provide users "anytime, anywhere" Internet services. Some perceived benefits [4], [8], [9] include enhanced resilience against node failures, channel errors and transient channel outages, higher data rates delivered to the users, and cost savings in tetherless deployment.

In this paper, we study the problem of resilient and high-throughput routing in a wireless mesh network. The two concrete goals for our routing design are: (1) resilience against lossy wireless links, transient/permanent channel outages, and occasional HAP node failures; (2) high data rate along the route that exploits receiver diversity (in terms of current perceived SNR) and the available multi-rate capability at the physical layer (e.g., all 802.11a/b/g/n devices possess this feature).

There are two fundamental challenges in routing over wireless mesh networks. First, routing design has to address issues in both short- and long-time scales. Similar to wired routing, coarse-grained routing maintains stable routes in the long term (e.g., tens of seconds or more). In the meantime, the fine-grained operation has to adapt to the instantaneous wireless channel variations (e.g., the channel coherence time is typically at the scale of a few milliseconds [3]) in order to achieve high throughput. A good wireless mesh routing algorithm has to both ensure long-term route stability and achieve short-term opportunistic performance. Second, wireless routing has to ensure robustness against a wide spectrum of soft and hard failures, ranging from transient channel outages, links with intermediate loss rates [16], to persistent channel disconnections, nodes under denial-of-service (DoS) attacks, and failing nodes. The state-of-the-art solutions do not address both issues. Moreover, they do not scale to large node population in scenarios such as city or metropolitan mesh networks.

In this paper, we describe ROMER, a resilient and opportunistic routing protocol for the emerging wireless mesh network. ROMER exploits the relatively dense deployment of HAP nodes and builds a randomized and opportunistic forwarding mesh to enhance robustness and throughput. In ROMER, each data packet carries a cost credit beyond the minimum required. The packet uses the carried credit to expand its traversed routes around the minimum-cost path to form a forwarding mesh on the fly. Within the runtime mesh, each intermediate node opportunistically selects the instantaneous, higher-quality wireless links to maximize the delivery throughput. The instantaneous routes are also randomized to minimize the forwarding overhead. This way, the per-packet forwarding mesh offers a set of *candidate* routes at coarse time scales to reflect long-term route stability. The actual forwarding path by each packet opportunistically adapts to the dynamic channel condition (in terms of current transmission rate) of each forwarding node, and the route adaptation is enabled at fine time scales (up to a few milliseconds of channel coherence time).

In summary, ROMER uses the credit mechanism to build a runtime forwarding mesh, centered around the long-term minimum-cost path between the source and the destination GAP, on a per-packet basis. The mesh provides inter-leaved forwarding paths to ensure high degree of robustness against various failures. It also enables intermediate nodes to adapt to the short-term, wireless channel dynamics by opportunistically delivering the packet along the highest-rate links. It avoids the performance penalty of repetitive retransmissions over persistently poor channel conditions along a single path, the scheme used by current routing protocols [5], [7], [12], [13], [14], [17] and 802.11 devices. This is achieved via exploiting path diversity of multiple routes.

Our simulations and analysis have confirmed the effectiveness of ROMER. The simulations show that ROMER can achieve about 68-195% higher throughput gain over single-path routing. Its robustness is also better than the single and multipath routing protocols. With a randomized forwarding probability set as 0.2, the successful packet delivery ratio of ROMER is about 92% in a 17-hop delivery path over 5% channel loss over each hop, whereas it is only 67% using 2-disjoint-path routing and a merely 42% using single-path routing. ROMER still delivers 20% more packets than the 2-path routing with a randomized forwarding probability of 0.2, when the node outage rate reaches 10%. ROMER also has provable performance against failures and channel variations.

The rest of the paper is organized as follows. Section II discusses routing issues in wireless mesh networks. Section III describes the ROMER design. Section IV provides simple performance analysis of ROMER. Section V evaluates its resilience and opportunistic throughput gain via simulations. Section VI compares ROMER with the related work, and Section VII concludes the paper.

II. ISSUES FOR ROUTING IN WIRELESS MESH NETWORKS

In a mesh network, multiple stationary HAPs form the wireless backbone and communicate with one another over multihop wireless channels [4], [8], [2]. Each HAP is typically equipped with several radios, and has accessed to infinite power supply (i.e., HAP is not battery powered, and energy efficiency is not a serious issue). The radio typically offers multi-rate options via adaptive modulations at the physical layer [1]. When the signal-to-interference ratio varies, the radio selects the highest possible transmission rate subject to a given BER. Among the radios configured at the HAP, one serves as the local AP for the client hosts in its cell. Each pair of neighboring HAPs communicates via another radio over an independent wireless channel to maximize system throughput. To improve spectrum efficiency, recent studies have motivated the use of sector antennas between two neighboring HAPs [4]; this configuration can effectively improve radio frequency reuse. In fact, it has been widely used in the wide-area cellular networks.

The above seemingly simple architecture offers several appealing features, as documented in the recent literature [8], [4]. First, it incurs low infrastructure cost. It not only gets rid of the wiring cost that hinders the fast deployment of the public hot-spots [4], but also allows for the use of unlicensed bands and incremental deployment. Mesh routers can combine low-cost radios with smart mesh software. Second, it ensures *robust* coverage. As calculated in [8], a 50-device mesh can offer about 40dB link gain advantage over the conventional point-to-multipoint (PMP) radio communication. The alternate paths offered by the rich mesh topology ensure robustness and reduce the need for high link margins. Third, mesh networks can offer broadband services. Multiple mesh hops typically increase the effective subscriber capacity, and 802.16a mesh mode scheduling also guarantees per-hop latency. Finally, a mesh network possesses appealing scaling property in its capacity. It provides inherently favorable signal-to-interference characteristics, and increasing subscriber density increases overall network capacity, as articulated by Dave Beyer in [8].

Within such a network infrastructure, routing over mesh networks needs to address two new issues:

- **Ensuring resilience**. Compared with the wired Internet, wireless routing has to cope with various types of expected and unexpected faults. The wireless channel is inherently error prone, and the loss rate can be quite high in a short period of time. Recent measurements [16] have shown that the loss is more or less uniformly distributed and links with intermediate loss rates are common. Moreover, nodes may fail unexpectedly or are under denial-of-service (DoS) attacks. In both scenarios, the routing protocol has to route around the problematic area. The conventional single-path routing is vulnerable to node or link failures. The multipath routing approach is better in terms of robustness, but it may not be responsive enough to cope with such channel and node failures. The reason is that most state-of-the-art multipath routing algorithms, in both wired and wireless settings, operate at the scale of 10s of seconds or longer. They tend to select the best stable routes in the long term.
  The popular approach to improving link reliability is through link-layer retransmissions (e.g., the typical 802.11 MAC offers up to seven retransmissions for lost packets). However, such an approach incurs high throughput penalty. Both analysis and measurements [3] have shown that the channel coherence time is at the millisecond granularity in typical settings. Therefore, lossy channel conditions typically persist for multiple packets. Retransmitting packets over the persistent, poor channels will only reduce the effective throughput. This has been observed in recent measurements [23].
- **Exploiting path diversity**. A key architectural benefit of mesh networks is to *skip around* obstacles, rather than blast over and through obstacles in the conventional PMP approach. Specifically, in an indoor mesh network, the path loss is typically driven by obstacles rather than distance. This leads to the Log-Normal path loss model [8], specified by $C + 10n\log_{10}(dist) + X_\sigma$, where X_σ is a random variable with standard deviation σ. In PMP networks, large σ is bad, since the design must accommodate for the worst case, e.g., leads to $\frac{1}{r^4}$ or $\frac{1}{r^5}$ models. In mesh networks, large σ, which indicates large path

diversity, is good. The best-case links are automatically selected and used. Even in the simplistic case, [8] has argued that mesh routing offers higher throughput. Using the 802.11a radios and assuming free-space path loss and common noise environment, [8] has shown that a direct path delivers 6Mpbs, while 9Mbps can be obtained over a two-hop path because shorter links will use 18Mbps due to 6dB less path loss. The multihop routing benefits even greater in non-free-space environment and/or when routing around obstacles. Therefore, how to opportunistically leverage the short-term path variations and diversity to maximize the end-to-end throughput becomes a key challenge. This requirement stipulates routing in wireless mesh networks has to adapt to transient channel dynamics at the millisecond time scale, in order to exploit the path diversity and achieve high-throughput delivery.

In this paper, we focus on routing data from a client machine, via a local HAP, to a destination host on the wired Internet. The main goal is to route from the local HAP to a destination GAP, in the presence of dynamic channel variations and various failures mentioned above. The solution can also be adapted to the case when a wired Internet host delivers data to a wireless client, using standard mobility support technique.

III. ROMER DESIGN

ROMER uses opportunistic, forwarding mesh adjusted on a packet basis to ensure robustness and high throughput. The mesh is centered around the long-term stable, minimum-cost path (e.g., the shortest path or long-term minimum-delay path), but opportunistically expands or shrinks at the runtime to exploit the highest-quality, best-rate links enabled by the physical-layer multirate options. The actual forwarding routes select the high-rate links out of the candidate routes offered by the mesh. The actual forwarding routes are also randomized to deliver redundant data copies in a controlled manner to ensure resiliency against lossy links and transient node outages. In short, ROMER takes a two-tier routing approach and balances between long-term optimality (e.g., in terms of hop count or average latency) and short-term opportunistic gain (in terms of path throughput). It has two components that work in concert for efficient performance tradeoff:

- *runtime candidate mesh*: a credit-based approach that allows each packet to build its forwarding mesh on the fly. The packet may follow a subset of the *candidate* interleaved paths offered by the mesh to resist against channel and node outages.
- *opportunistic and randomized forwarding on the mesh*: to maximize the end-to-end throughput, ROMER uses greedy forwarding to opportunistically deliver the data packet along the instantaneously highest-rate link with probability one and other high-rate downstream links with high probability.

Compared with the popular single-path routing protocol in mesh networks, ROMER enhances both resilience and performance. Current protocols rely on link-layer retransmissions to recover from link loss. It thus suffers from repetitive retransmissions over persistent, poor links. Instead, ROMER exploits path diversity by transmitting to multiple receivers, at least one of which is more likely to be in good channel condition. This way, ROMER may significantly reduce the throughput overhead due to retransmissions. Leveraging path diversity also enables ROMER to opportunistically select the highest-rate link at the moment. In contrast, current protocols cannot adapt at the millisecond time scale since its routing metrics are updated every 10s of seconds. This also incurs throughput penalty by hindering the adaptivity of the routing protocol.

We next describe each component of ROMER in details.

A. Building Candidate Forwarding Mesh on the Fly

Given a source HAP (that connects to the client) and a destination/sink GAP, instead of single-path or multiple-disjoint-path routing, ROMER builds a forwarding mesh on the fly. It is well known that single-path delivery is prone to random node failures and channel losses. Retransmissions help but incur throughput penalty. Instead, the mesh is built around the minimum cost (in terms of long-term average delay or hop count), which reflects the long-term optimality. Moreover, the mesh provides enough flexibility of rich, interleaved paths to accommodate the short-term channel dynamics and transient outages. It allows for optimizing short-term, opportunistic performance regarding channel variations, while bounding the maximum deviation from the long-term optimal path.

In the following, we describe a novel credit-based approach to constructing a runtime forwarding mesh, which achieves flexible tradeoff between robustness, opportunistic gain and cost by controlling the credit carried by the data packets. We assume that each HAP records its minimum cost to each of the few GAPs. This minimum cost can be obtained via simple flooding or other available protocols, e.g., the one described in [22].

1) Overview: Intuitively, if a data packet carries an “extra” amount of credit cost beyond C_{source}, the minimum required cost from the source to the destination, the packet can afford to travel more paths during the delivery process. These paths interleave and form a candidate forwarding mesh, as illustrated in Figure 1). Upon receiving a packet, a node checks whether the packet has enough credit for the node to further forward downstream. If so, the node uses opportunistic and randomized forwarding (to be described in Section III.B) to deliver the packet. Otherwise, it stops forwarding the packet. Obviously, the more extra credit a packet has, the wider the mesh can be. Therefore, we can easily adjust the width of the forwarding mesh on a per-packet basis, by controlling the extra credit granted to the packet.

In order for the above seemingly simple design to work, we need to address two critical issues: (1) How to distribute the credit along multiple, intermediate hops? (2) How to control the overhead due to delivery of redundant copies of the packet on the mesh?

In order to address the issue of credit distribution, we first note that, not all paths satisfying the total budget requirement

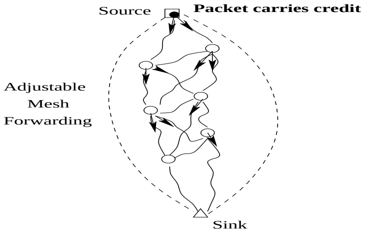

Fig. 1. Credit-based runtime mesh

(i.e., C_{source} plus the total credit) should be used in forwarding. For example, given a credit of 50 and the minimum cost at the source of 100, the packet may go along a particular path with cost 150. However, there is little chance such a delivery could succeed, in the presence of channel errors and node failures. Because there may be a large number of intermediate nodes between the source and the destination GAP, channel error over any hop could ruin the delivery. Similarly, a path with cost 140 might be a bad choice as well. Packet forwarding along such paths makes marginal contributions to successful delivery. Fundamentally, these paths do not have sufficient remaining credits to successfully deliver the packet to the destination GAP.

Therefore, we should confine the mesh to the "good" paths, which have enough remaining credits to ensure successful delivery to the destination. This is related to how to distribute the overall credit over each hop from the source to the sink. Without any control over credit distribution, the first several hops can consume most credit, and eventually all downstream hops end up with single-path forwarding due to insufficient amount of credit line. Therefore, each hop should not use excessive amount of credit to avoid stress downstream-hops' credit.

There may be different design choices for credit distribution over each hop from the source to the destination. For example, one may chose to give more credit to hops in areas of stronger interference. One may also give slightly more credits to upstream nodes. In this work, Given the dense deployment of mesh nodes, we explore a policy that allocates more credit for the beginning hops in order to quickly expand the mesh, and less at later hops after the mesh is already wide enough.

Specifically, we want the credit received by a hop to be in proportion to how "far" it is to the destination. This policy can achieve good robustness if channel error and node outages follow similar statistical distributions across different HAPs in a long run.

Note that the construction of the forwarding mesh is completely *data driven*. Each data packet can specify its own credit line (at the source) to satisfy its customized robustness requirement. This also provides a nice mechanism to provide *prioritized/differentiated robustness* for different categories of data packets.

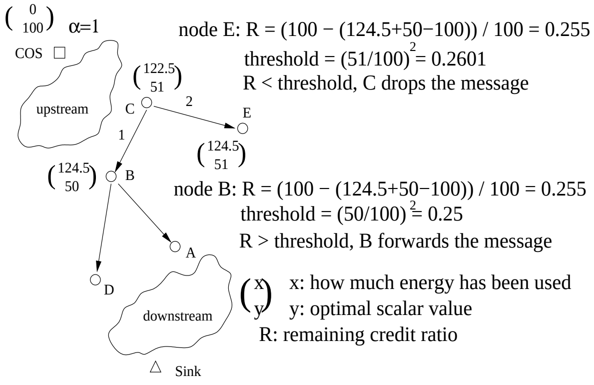

Fig. 2. Mesh-based forwarding example

To address the second issue of confining the forwarding overhead, we use the randomization technique to prune the runtime mesh. Each intermediate node probabilistically forwards the packet to eligible downstream nodes that satisfy the local credit requirement.

2) Algorithm: We now present the algorithm that realizes the above policy. At the source, the credit is set as $\alpha \cdot C_{source}$, where α is a constant, and C_{source} is the minimum cost from the source to the destination GAP. A packet carries the total budget $(1+\alpha) \cdot C_{source}$, and its current cost expenditure, denoted by P_a, along its way to the destination GAP.

We use the example of Figure 2 to explain the algorithm. When the source (in the upper-left corner) sends out a message, it specifies a credit limit of 100 units beyond its cost 100 ($\alpha = 1.0$). Thus the packet may consume 200 units of cost (say, energy) as it traverses from the source to the sink GAP. At an intermediate node C, assume that the packet has consumed cost 122.5 units from the source to C. Now C broadcasts the packet consuming 2 units of cost, reaching both B and E. E performs the following calculations:

- If E were to forward this packet along its single path, the total amount of cost needed would be $122.5 + 2 + 51 = 175.5$, of which $122.5 + 2$ is the amount that has been used, and 51 is E's cost.
- The amount of credit needed is $175.5 - 100 = 75.5$, where 100 is the source's cost. Given a credit budget of 100 carried by the packet initially, the remaining credit available for E and downstream nodes is $100 - 75.5 = 24.5$. The ratio of remaining credit to initial credit line is $24.5/100 = 0.245$.
- Now we compare the above remaining credit ratio 0.245 to an threshold value, which is $(51/100)^2 = 0.2601$, where 51 is E's cost, and 100 is the cost of the source. E decides to silently discard the packet since the remaining credit ratio 0.245 is less than the threshold 0.2601.

Similar calculations are carried out by node B, which leads to a remaining credit ratio of 0.255 and a threshold of 0.25 (refer to Figure 2 for detailed calculations). Since the remaining credit ratio is greater than the threshold, B will forward the message with probability p.

Now we explain how the above comparison against the threshold leads to approximately linear credit distribution from the source HAP to the destination GAP. That is, the credit received by a hop is proportional to its cost to the GAP. At the source (with cost C_{source}), credit $\alpha \cdot C_{source}$ is carried by the packet. The packet has consumed cost P_a when it reaches node A that has cost C_A. At node A, maximum credit is consumed when the remaining credit ratio α_A is equal to the specified threshold $\alpha(A)$. It is easy to see that $\alpha_A = \frac{\alpha \cdot C_{source} - (P_a + C_A - C_{source})}{\alpha \cdot C_{source}}$, and $\alpha(A) = \left(\frac{C_A}{C_{source}}\right)^2$. Therefore, we have: $\frac{\alpha \cdot C_{source} - (P_a + C_A - C_{source})}{\alpha \cdot C_{source}} = \left(\frac{C_A}{C_{source}}\right)^2$. Then it follows that $P_a = -C_A - \alpha \cdot \frac{C_A^2}{C_{source}} + (1+\alpha)C_{source}$. Taking the derivative, we arrive at $\frac{\partial P_a}{\partial C_A} = -\left(1 + 2\alpha \frac{C_A}{C_{source}}\right)$. Then, the allowed cost consumption at a hop is:

$$\Delta P_a = -\Delta C_A - 2\alpha \frac{C_A}{C_{source}} \Delta C_A$$

In the above expression, the first term denotes the minimum required cost to go to the next hop. The second term is the amount of credit that can be used locally (without overspending the credit for later hops); it is proportional to C_A, the cost from this node to the GAP. Therefore, as a packet traverses from the source to the GAP, it is allowed to consume more credit near the source and less credit near the destination. This way, the forwarding mesh will be expanded more aggressively initially, and less aggressively downstream.

In addition to the randomized forwarding to control excessively redundant packet deliveries, we may also suppress duplicate copies to further reduce the overhead when the wireless channel quality is good. Note that a node may receive multiple copies of the same packet, each of which passes the threshold comparison (i.e., the packet has sufficient remaining credit). To save forwarding cost, each intermediate HAP maintains a cache that stores the sequence numbers of its recently forwarded packets. When a node receives a packet, it compares the sequence number against those in the cache. The packet can be discarded if it is a duplicate; otherwise, the cache is updated with the new packet. Therefore, each node can forward the packet once only once in the extreme case.

In summary, the credit-based approach allows each packet to carry dynamic credit state to use different meshes on the fly and meet its robustness requirement. This is achieved without modifying the operations at each HAP node, in sharp contrast to early multipath routing that requires infrastructure maintenance. There is no need to maintain extra state information on the mesh at each HAP. A HAP node knows whether it belongs to the mesh only *after* it receives the packet. Moreover, mesh provides richer connectivity than multiple, explicit, forwarding paths. In the presence of channel and node outages, no effort is needed to actively repair the broken paths as in multipath routing. In addition, carrying state in each packet helps the design to scale to large mesh network size deployed in metropolitan or large corporate settings. The algorithm also involves a few simple operations, and the computation complexity makes it scale to high-speed forwarding at each HAP. Furthermore, no routing loops can form in the forwarding process. When receiving a packet, the receiver compares its cost to its sender's cost to ensure it is in the decreasing direction on cost. The packet is always delivered downstream.

B. Randomized Opportunistic Forwarding

The goal of opportunistic forwarding is to leverage the short-term channel diversity to select the highest-throughput link for packet delivery on the runtime mesh. Wireless channel quality may vary greatly in the short term due to multipath fading, obstacles, mobile objects, interferences and environmental noise. Therefore, the perceived signal-to-interference ratio (SNIR) will change over time. The multirate option offered by adaptive modulations dynamically adjusts a node's transmission speed to match the current SNIR. In practice, the transmission rate of an 802.11a/g radio can vary from 6Mbps up to 54Mbps in an indoor environment.

The opportunistic forwarding in ROMER takes a simple, greedy approach. It exploits the dense node deployment in terms of downstream neighbors. Consider each node has at least K_n neighbors. Once the mesh is constructed for a data packet using the credit mechanism, each intermediate node selects with probability 1 the best downstream link, which offers highest instantaneous rate, to maximize the opportunistic throughput gain. For other eligible downstream links that fit within the local credit bound, ROMER also favors better-quality links over low-quality links by selecting higher random forwarding probability p. In the current design, the forwarding probability p is set to be in proportion to each link's current transmission rate, normalized with the best downstream link speed. Therefore, given the current rate of link l as R_l, the forwarding probability over this link is set as $p_l = \frac{R_l}{R_{\max}}$, where $R_{\max}$ is the highest-rate link among all downstream links. To further control the overhead, we bound the total number of forwarded copies for a packet, say K. Then, each link's forwarding probability is set as $p_l = \frac{R_l}{\sum_i R_i} \cdot K$. The number of forwarded copies is a function of the link loss probability. Section 4 provides a heuristic calculation.

The opportunistic forwarding in ROMER leverages the available MAC-layer mechanisms to estimate the transmission rate to each downstream node. ROMER can work with any rate adaptation mechanism, e.g., Auto Rate Fallback (ARF) or RBAF [10] in the MAC. Similar to RBAF, it can use the RTS/CTS handshake to let the receiver choose the best rate for data transmissions. The forwarding probability can also be negotiated and adjusted via RTS/CTS according to the current channel condition. Fortunately, the rate estimation does not need to be done on a per-packet basis, which consequently incurs high overhead. This is because the channel coherence time is typically a few milliseconds or longer.

IV. PERFORMANCE ANALYSIS

In this section, we provide simple analysis of the opportunistic throughput gain and robustness of ROMER.

A. Opportunistic Throughput Gain

In the following analysis, we show that the simple greedy forwarding policy, in which an intermediate HAP forwards the packet along the highest-data-rate link, can deliver near-optimal end-to-end throughput under certain conditions.

We assume that the rates of different links vary independently, following a general discrete probability distribution. Specifically, there exist q possible data rates, $r_1 \leq r_2 \leq \cdots \leq r_q$, which are used with a probability of $p_1, \cdots, p_q$, respectively. We consider the end-to-end throughput of a h-hop forwarding path, denoted by $V_1 \rightarrow V_2 \rightarrow \cdots \rightarrow V_h$, that is determined by the greedy forwarding algorithm. Let k_i be the number of neighbors that node V_i has. The metric of interest is the *competitive ratio*, defined as the ratio between the expected link-layer throughput of the greedy algorithm and that of a globally optimal algorithm. We have the following theorem that characterizes the lower bound for such competitive ratio:

Theorem 4.1: The competitive ratio of the greedy algorithm is lower bounded by

$$\rho \geq \sum_{j=1}^{q} \frac{r_j \gamma(j,k)}{r_q} \tag{1}$$

where $k = \min(k_1, \cdots, k_{h-1})$, and $\gamma(j,k) \triangleq \sum_{i=1}^{k} p_j (\sum_{l=1}^{j-1} p_l)^{i-1} (\sum_{l=1}^{j} p_l)^{k-i}$ □

PROOF: The end-to-end throughput T is the minimum of the per-link throughput along the forwarding path, that is, $T = \min(T_1, T_2, \cdots, T_{h-1})$, in which T_i is the throughput of the link between node V_i and node V_{i+1}. Based on Lemma 4.1 that is described next, we know that the expected per-link throughput is: $T_i = \sum_{j=1}^{q} r_j \gamma(j, k_i)$. □

The proof of the above theorem uses the following Lemma:

Lemma 4.1: Let $x_1, \cdots, x_k$ be k independent random variables that follow the same distribution(i.i.d.) as below

$$P[x_i = r_j] = p_j \quad j = 1, \cdots, q$$

in which $r_1, \cdots, r_q$ are q different values. We have

$$E[\max(x_1, \cdots, x_k)] = \sum_{j=1}^{q} r_j \gamma(j,k) \tag{2}$$

where $\gamma(j,k) \triangleq \sum_{i=1}^{k} p_j (\sum_{l=1}^{j-1} p_l)^{i-1} (\sum_{l=1}^{j} p_l)^{k-i}$

PROOF: By definition, we have:

$$E[\max(x_1, \cdots)] = \sum_{j_1=1}^{q} \cdots \sum_{j_k=1}^{q} max(r_{j_1}, \cdots, r_{j_k}) p_{j_1} \cdots p_{j_k}$$

In order to solve the above equation, we divide the k-dimensional space into k sub-regions. In sub-region i, x_i is the first maximum in the array. That is, x_i is strictly larger than $x_1, \cdots, x_{i-1}$, and no less than $x_{i+1}, \cdots, x_k$. Thus, we can rewrite the above equation as:

$$\begin{aligned} & E[\max(x_1, \cdots, x_k)] \\ = & \sum_{i=1}^{k} \sum_{j=1}^{q} r_j p_j (\sum_{l=1}^{j-1} p_l)^{i-1} (\sum_{l=1}^{j} p_l)^{k-i} = \sum_{j=1}^{q} r_j \gamma(j,k) \end{aligned}$$

# of neighbors	Competitive Ratio
2	0.6554
4	0.8213
6	0.8953
8	0.9339
10	0.9561

TABLE I

COMPETITIVE RATIOS OF GREEDY FORWARDING

Given that the function $\gamma(j, \beta)$ increases monotonically with respect to β, we can see that $T = \sum_{j=1}^{q} r_j \gamma(j,k)$, where $k = \min(k_1, \cdots, k_{h-1})$. The lower bound of the competitive ratio can be easily seen because the end-to-end throughput of a global optimal algorithm is at most r_q. □

As a special case, when the date rates are uniformly distributed, i.e., $p_i = \frac{1}{q}$, we can derive the following corollary from Theorem 4.1:

Corollary 4.1: With uniform distribution of data rates, the The competitive ratio of the greedy algorithm is lower bounded by:

$$\rho \geq \frac{1}{q^k r_q} \sum_{i=1}^{k} \sum_{j=1}^{q} r_j j^{k-i} (j-1)^{i-1} \tag{3}$$

where $k = \min(k_1, \cdots, k_{h-1})$. □

In fact, the above competitive ratio is approaching 1 in the realistic scenarios. In other words, the greedy algorithm in ROMER can deliver near-optimal end-to-end throughput. For example, in the 802.11a/g networks, there are 8 possible data rates: 6, 9, 12, 18, 24, 36, 48, and 54 Mbps. We plug these parameters into Equation 3, and vary the minimum number of neighbors, k, from 2 to 10. The competitive ratios are shown in Table I. When the network is reasonably dense, say each node has 6 neighbors on average, the throughput achieved by greedy forwarding algorithm is 89.53% of that yielded by the global optimal algorithm.

While our analysis is based on the link-layer throughput, it can be easily extended for the *effective* throughput with lossy channels. In particular, with independent channel loss, the greedy algorithm can also deliver near-optimal effective throughput in the network layer (the proof is similar to Theorem 4.1).

B. Robustness

Another important metric of routing performance is the end-to-end *delivery ratio*, defined as the percentage of packets that successfully reach the destination. Next we show that the

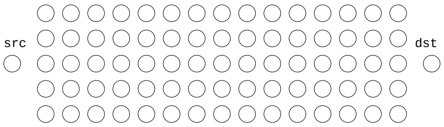

Fig. 3. Simulations topology

per-hop K-redundant forwarding in ROMER can achieve a constant end-to-end delivery ratio.

We consider a simplified scenario in which each link has an independent loss rate of P_l, and the end-to-end path has h hops. The required end-to-end delivery ratio is a constant α. It is easy to see that the delivery ratio over a single path is $D_s = (1-P_l)^h$. Because ROMER delivers the packets along multiple disjoint paths, the number of which is denoted as M, the end-to-end delivery ratio is improved to $D = 1-(1-D_s)^M$. Therefore, we have $1-(1-D_s)^M = \alpha$. Solving this equation yields

$$M = (\frac{1}{1-P_l})^h \times \ln\frac{1}{1-\alpha} \quad (4)$$

That is, to achieve a constant end-to-end delivery ratio, the number of disjoint paths should increase exponentially with respect to the path length. This is why each HAP forwards the packet to K different neighbors. In fact, based on Equation 4 we know that K should be set to $\frac{1}{1-P_l}$.

V. SIMULATION

In this section, we evaluate the performance of ROMER by simulations. We first quantify the throughput gain of ROMER at different forwarding probabilities. We also study the successful packet delivery ratio at the destination GAP, at different link and node failure percentages. These results show that ROMER can achieve about 68-195% higher throughput gain over single-path routing. The successful packet delivery ratio of ROMER is about 92% in a 17-hop delivery path over 5% channel loss over each hop, whereas it is only 67% using 2-disjoint-path routing and a merely 42% using single-path routing.

We use *ns-2* to carry out our simulations. A representative simulation topology is shown in Figure 3. In this topology, the source HAP and the destination GAP are 17 hops away, and the source generates constant-bit-rate traffic to the destination. We assume each TAP node can communicate with all its neighbors by sector antennas using 802.11g/a radios. The data rate varies between 6Mbps and 54Mbps, depending on the channel fading condition. The Ricean Fading model implemented by Rice University is used to simulate channel fading. The single-path distance-vector routing algorithm and its simple modified 2-disjoint-path routing are used for comparisons. Ideally, we'd like to use more sophisticated multipath routing algorithms for thorough performance comparison in the simulations, but no such a simulator is available at this time.

A. Throughput Gain

In this simulation, we evaluate the throughput performance of ROMER. Figure 4 shows the throughput gain of ROMER over that of single-path routing, for various link loss percentages and different forwarding probability p by each HAP. We observe that, the end-to-end throughput of ROMER increase by 68-195%, when the forwarding probability is 0.1-0.2 at each HAP node, and the channel error rate is 0.1-5%. Moreover, the throughput gain grows as the channel loss percentage increases.

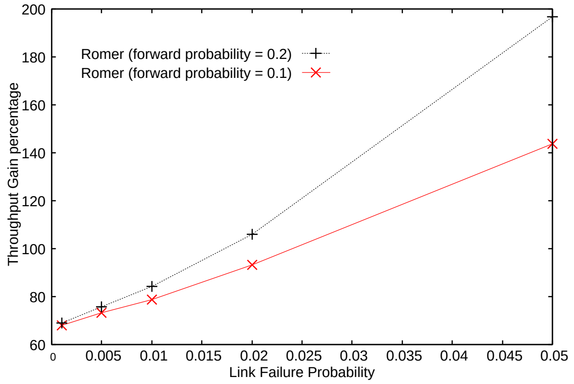

Fig. 4. Throughput gain of ROMER compared with single path routing

There are two reasons why ROMER achieves higher end-to-end throughput. First, it is because of the opportunistic forwarding mechanism built in ROMER. ROMER opportunistically favors the paths with higher data rates. As the data rate of the channel varies over time due to short-term fading or obstacles, most single path routing algorithms remain oblivious and will suffer because one link over the multihop route may happen to be at low data rate. In ROMER, however, routing always forwards the data packet to the downstream node with highest data rate. Second, ROMER provides more robust transmissions (to be shown in Section V-B). Both factors of opportunistic forwarding and robust delivery contribute to the throughput gain in the presence of time-varying transmission rates and channel losses.

B. Resilience Against Lossy Links and Node Outages

In this section, we study the robustness of ROMER against channel loss and node outage. We let channel error and node failures vary following uniform random distributions in all simulations. The successful delivery ratio is defined as the percentage of packets that have arrived at the destination GAP, out of all client packets sent out by the source HAP.

Figure 5 plots the successful delivery ratio of ROMER at different channel error percentages. From the figure, we see that ROMER can achieve about 90% or higher packet delivery ratio with a forwarding probability 0.2, when the channel error rate varies from 0.1% to 5%. In contrast, the delivery ratio by the single-path routing drops from 98% to 42% when the channel loss reaches 5%. Even for the 2-disjoint-path routing, the delivery ratio is only 67% at 5% channel loss. In all cases, ROMER clearly outperforms the conventional single and multipath routing protocols.

Figure 6 plots the successful delivery ratio of ROMER,

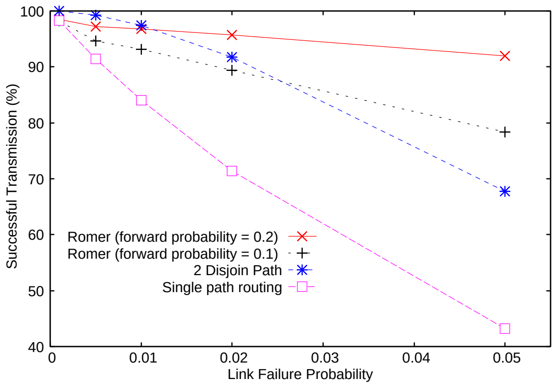

Fig. 5. Successful delivery ratios by ROMER, single-path routing, and two-disjoint-path routing at different channel error percentage

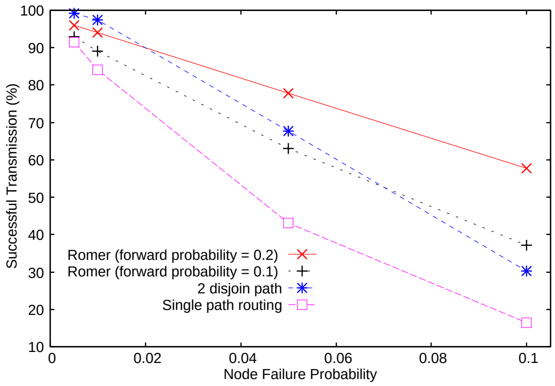

Fig. 6. Successful delivery ratios by ROMER, single-path routing, and two-disjoint-path routing at different node failure rate

single-path routing and 2-path routing at different node failure rates. The figure shows that, ROMER still achieves about 80% delivery ratio at a randomized forwarding probability of 0.2 when the node failure rate is 5%, whereas the ratio is 70% for 2-path routing, and 42% for single-path routing. Even when the node outage rate increases to 10%, ROMER still delivers 20% more packets than the 2-path routing, at a small randomized forwarding probability of 0.2.

In multihop routing over wireless mesh networks, if the number of hops on the end-to-end path is large (e.g., more than 15 hops), the successful packet delivery ratio will be low even with small channel/node failure rate. As a result, single-path routing and multipath routing will not offer maximum degree of robustness. One the other hand, in ROMER, in addition to forward the packet to the highest-rate downstream node, each node also forwards packets to other downstream nodes with a small forwarding probability. These extra copies of packets (say, 1.3 copies on average) can greatly enhance the end-to-end successful delivery ratio, in the presence of transient/persistent channel loss and node outage due to failures or under DoS attacks.

VI. RELATED WORK

Routing has been a very active research area in the context of ad hoc networks, and many proposals have appeared in the literature (see [12], [13], [14] for a few samples). However, the scale of ad hoc networks is typically much smaller compared with mesh networks, and these proposals typically assume a much smaller network size (e.g., DSR assumes a network size of 6-8 hops [12]). Moreover, they mainly focus on the mobility, whereas our focus is on resilience and high data throughput. In general, these routing protocols function at the coarse time scale (several seconds or more), whereas ROMER operates at both coarse time scale (i.e., the credit is updated every a few seconds) and fine time scale (i.e., actual path is selected at the per-packet level) to exploit the opportunistic channel gain.

In recent years, there have been an increasing number of studies on wireless mesh networks [2], [4], [5], [8], [9], [15], [16], [17], [18]. However, ROMER addresses the different problem of resilient and high-throughput routing compared with the work on mesh networks in the literature. [4], [8], [9], [18] articulated the architectural benefits of mesh networks compared with the conventional cellular or mobile ad-hoc networks. [2] seeks to define the MAC standard, and [15] discusses link-layer scheduling and end-to-end fairness issue. [16] provides a detailed link-level measurement study on an 802.11b mesh network. [5] addresses the problem of high-throughput routing. It design of Multi-Radio Link-Quality Source Routing (MR-LQSR) is a source-routing protocol, but with a new path metric called weighted cumulative expected transmission time (WCETT) that reflects the loss rate and link bandwidth. The metric update is done at time scale much slower than the packet level. ROMER addresses a general routing problem of resiliency and high rate. Its two techniques of building a runtime mesh and exploiting opportunistic and randomized forwarding on the mesh are also significantly different from [5].

The design of ROMER – runtime forwarding mesh, randomized and opportunistic delivery on the mesh – bears conceptual similarity with the related work but has fundamental differences. Forwarding meshes were used in [19], [20], [21] to enhance robust multicast delivery in wireless ad hoc networks. However, these proposals require intermediate nodes to maintain explicit states about whether they are in a mesh. Control messages are exchanged in the network to maintain meshes. In ROMER, the mesh is formed on the fly for each packet. Besides, we can readily adjust the width of the mesh by changing the credit line at the source. [11] also constructs a forwarding mesh on the fly but in a different context of large sensor networks. The goals are to improve protocol scaling and robustness. It did not make any efforts to optimize the transient throughput as in ROMER. Moreover, the overhead in [11] is more significant since it did not have the notion of the randomized forwarding. In another work, [7], [17] also applies the concept of opportunistic routing to multihop wireless routing. However, they seek to decrease the total number of transmissions. The design exploits the feature of probabilistically independent reception of transmissions at different nodes, including high-loss long-distance links. It does not explicitly leverages the multirate option at the physical layer. In ROMER, we leverage the transient transmission rate variations enabled by the multirate capability and select the highest throughput path, instead of the closest receiver to the destination as in [7]. We also address the issue of resilience while [7] does not. The notion of randomized forwarding has also been explored in a different context of resilient application-layer wired multicast [6]. However, this concept is used together with the opportunistic forwarding in ROMER. We tune the forwarding probability to each link's instantaneous throughput. Moreover, we have dual goals of improving resiliency and maximizing throughput, where [6] focuses on resiliency on wired multicast only.

VII. CONCLUSION

In this paper, we describe ROMER, a resilient and high-throughput routing protocol for wireless mesh networks. ROMER takes a two-tier routing approach and balances between long-term stable paths and short-term opportunistic detours. It both ensures robustness against transient/persistent link loss and node outages, and exploits path diversity to maximize the end-to-end throughput.

ROMER has exploited two novel techniques in its design. It constructs a runtime forwarding mesh on a per packet basis with negligible effort by the infrastructure, using a new credit-based mechanism. Each packet can specify its own credit based on its individual resilience requirement. The mesh is formed on the fly as the packet moves toward the destination. ROMER also explores opportunistic forwarding to deliver the packet along the highest-transmission-rate paths. A simple greedy algorithm can deliver near-optimal throughput. To control the overhead, ROMER also uses randomized forwarding at each intermediate node. Our simulations and analysis have shown that ROMER is able to achieve up to 195% throughput gain compared with the conventional single-path routing. Its packet delivery ratio can be 40% more compared with the single-path and two-disjoint-path routing protocols, when the channel loss is about 5% and the node failure is about 10%. This performance is achieved with a randomized forwarding probability of 0.2 in the simulations.

Ongoing work on ROMER seeks to refine the analysis, work out the security aspect to enhance resilience against malicious routing attacks, and add mobility support component to handle client mobility.

REFERENCES

[1] B. O'Hara and A. Petrick. IEEE 802.11 Handbook, A Designer's Companion. IEEE Press, 1999.

[2] IEEE 802.16a-2003 IEEE Standard for Local and metropolitan area networks.

[3] Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, Opportunistic media access for multirate ad hoc networks, in MOBICOM 2002.

[4] R. Karrer, A. Sabharwal, and E. Knightly, Enabling Large-scale Wireless Broadband: The Case for TAPs, HotNets 2003.

[5] Richard Draves Jitendra Padhye Brian Zill, Routing in multi-radio, multi-hop wireless mesh networks, in MOBICOM 2004.

[6] Suman Banerjee, Seungjoon Lee, Bobby Bhattacharjee, Aravind Srinivasan, Resilient Multicast using Overlays, ACM SIGMETRICS 2003.

[7] Sanjit Biswas,Robert Morris,Opportunistic routing in multi-hop wireless networks, ACM SIGCOMM Computer Communication Review, 2004

[8] Dave Beyer, Wireless Mesh Networks for Residential Broadband, National wireless Engineering conference, 2002.

[9] Wireless Mesh Network Projects at Intel. online document.

[10] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multi-hop wireless networks. In MOBICOM 2001.

[11] Fan Ye, Gary Zhong, Songwu Lu, Lixia Zhang, GRAdient Broadcast: A Robust Data Delivery Protocol for Large Scale Sensor Networks, to appear in ACM Wireless Networks (WINET), Vol. 11, No.2, March 2005.

[12] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-hoc Wireless Networks," *Mobile Computing*, Kluwer Academic Publishers, 1996.

[13] C. Perkins et al., "Ad-Hoc On Demand Distance Vector Routing (AODV)," Internet-Draft, 1997.

[14] J. Broch, D. A. Maltz, D. Johnson, Y. Hu, J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *MOBICOM'98*.

[15] V. Gambiroza, B. Sadeghi, and E. Knightly, "End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks," in ACM MobiCom 2004.

[16] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, Robert Morris, "Link-level Measurements from an 802.11b Mesh Network," SIGCOMM 2004.

[17] S. Biswas, R. Morris, ExOR: Opportunistic Multihop Routing for Wireless Networks, SIGCOMM 2005.

[18] J. Bicket, D. Aguayo, S. Biswas, R. Morris, Architecture and Evaluation of an Unplanned 802.11b Mesh Network, MOBICOM 2005.

[19] S.-J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks," Internet Draft, draft-ietf-manet-odmrp-02.txt, 2000.

[20] Ching-Chuan Chiang, Mario Gerla, "Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks," Cluster Computing, 1998.

[21] J.J. Garcia-Luna-Aceves and E.L. Madruga, "Core assisted mesh protocol," IEEE JSAC, Aug. 1999.

[22] Fan Ye, Alvin Chen, Songwu Lu, Lixia Zhang, "A Scalable Solution to Minimum Cost Forwarding in Large Scale Sensor Networks." ICCCN 2001.

[23] P. Zerfos, G. Zhong, J. Cheng, H. Luo, S. Lu, J. Li, DIRAC: A Software-based Wireless Router, MOBICOM 2003.