Routing with Uncertainty in Wireless Mesh Networks

Fajun Chen*, Jiangchuan Liu†, Zongpeng Li‡, Yijie Wang*
* School of Computer Science, Univ. of Defense Technology, China
† School of Computing Science, Simon Fraser University
‡ Department of Computer Science, University of Caglary

***Abstract*—Existing routing protocols for Wireless Mesh Networks (WMNs) are generally optimized with statistical link measures, without focusing on the intrinsic uncertainty of wireless links. We show evidence that, with the transient link uncertainties at PHY and MAC layers, a pseudo-deterministic routing protocol that relies on average or historic statistics can hardly explore the full potentials of a multi-hop wireless mesh.**

We study optimal WMN routing using probing-based anypath forwarding, with explicit consideration of transient link uncertainties. Starting from a two-state link capacity model, we show the underlying connection between WMN routing and the classic Canadian Traveller Problem (CTP) [1]. Inspired by a stochastic recoverable version of CTP (SRCTP), we develop an practical SRCTP-based online routing algorithm under link uncertainties. We study how dynamic next hop selection can be done with low cost, and derive a systematic selection order for minimizing transmission delay. We further extend our solution to a general multi-rate link model, and present a Stopping Theory (ST) based solution, which naturally degrades to the SRCTP algorithm in the two-state link model. We conduct simulation studies to verify the effectiveness of the SRCTP and ST algorithms under diverse network configurations. In particular, compared to deterministic routing, reduction of end-to-end delay ($51.15{\sim}73.02\%$ for two-state links, 5.16% for multi-rate links) and improvement on packet delivery ratio (99.76% for two-state links, 94.44% for multi-rate links) are observed.

I. Introduction

A Wireless Mesh Networks (WMNs) consists of backbone wireless routers that are relatively static and resource-rich, as well as wireless end-terminals that communicate through multi-hop routing, with possible Internet connection via the backbone routers. Given its cost-effectiveness in rapidly bringing a large number of users online, WMNs have enjoyed rapid growth in deployment during the past decade [2]. However, WMN routing has been notably challenging, despite much smaller network sizes than that of the Internet. Wireless links are intrinsically unreliable, at both long (session) and short (packet) time scales. The physical layer modulation switches dynamically, interference happens frequently, and effective link capacities fluctuate from time to time. All these are further complicated by the multi-hop relay of packets.

Given the uniqueness of WMNs, a series of routing protocols have been proposed, addressing link unreliability from various aspects [3]–[6]. While they are generally more reactive to link dynamics than Internet routing protocols, the elastics remain in a relatively long time scale. These routing algorithms strike to discover an optimal route and periodically update it, often in a relatively high frequency or per session demands. However, once determined, every relay node will have a fixed next hop for forwarding data. Better flexibility can be enabled by multi-path routing, but the next-hop selection for each individual packet is still largely based on pre-assignment or historic statistics [7], [8].

This convention however does not well-address the intrinsic uncertainty of wireless links, particularly in multi-hop routing. Intuitively, a link that statistically performs well on average may be unreliable and suffer from intermittent failures. A fixed route containing such a link consequently witnesses packet losses and long packet delays from time to time. During the intermittent failures, other links may be available with better performance. Such link availability and status information are discovered only in an *online* fashion, *i.e.*, at the time of transmission. In other words, with the transient link uncertainties at PHY and MAC layers, a (pseudo) deterministic routing protocol that relies on average or historic statistics can hardly explore the full potentials of a wireless mesh.

Recently, opportunistic routing (OR) was proposed [9], [10] to cope with the unreliability of wireless links. OR exploits the broadcast nature of wireless transmission, and runs a forwarder selection procedure to decide which node in the neighborhood of the transmitter further relays the packet. While leading to improved throughput, OR protocols are also known to incur control overhead due to the forwarder selection at each hop, contributing to prolonged end-to-end packet delays.

We propose to study delay-optimal routing in WMNs using a probing based anypath forwarding approach, with explicit consideration of transient link uncertainties. We start with a simple two-state model, where each link either fails or works at full rate. We establish the underlying connection between WMN routing and a Stochastic Recoverable version of the classic Canadian Traveller Problem (SRCTP) [1], and develop an SRCTP-based online routing algorithm for delay minimization. As a key operation of this stochastic algorithm, a node performs online probing of its neighbors' availability upon each packet transmission, and determines its next hop on the fly accordingly. We study how such probes can be done with low cost, and derive a systematic probing scheme for delay minimization.

We then extend our study of the stochastic routing algorithm to a more general multi-rate link model. Now that links can have any real-time capacity between zero and the full rate, neighbor probing and selection become more challenging. In particular, when a probed neighbor is currently working at an intermediate rate, it is unclear whether we should pick up this neighbor immediately as in SRCTP, or keep probing for a potentially better one. We demonstrate that this generic problem model can be solved by applying *Stopping Theory*

(ST) [11], and present an ST algorithm based on a threshold that is optimally updated with the routing table.

An interesting observation is that ST naturally degrades to SRCTP in the two-state link model, although we have designed these two algorithms from different perspectives, using different theoretical tools. Such a unification is in line with the fact that both algorithms achieve optimal expected routing delay. Through extensive simulations, we demonstrate the effectiveness of SRCTP and ST under various network configurations. In particular, we observed considerable reduction on end-to-end delay ($51.15 \sim 73.02\%$ for two-state links, 5.16% for multi-rate links) and improvement on packet delivery ratio (99.76% for two-state links, 94.44% for multi-rate links), as compared to traditional WMN routing.

The rest of the paper is organized as follows. Sec. II presents related research, and Sec. III describes the network model. Sec. IV presents the SRCTP online routing algorithm for the two-state link model, then Sec. V extends to the multi-rate link model. Section VI shows the simulation results, and Section VII concludes the paper.

II. Previsous Research

Wireless transmissions are susceptible to spatial interference, and routing in multi-hop wireless networks faces significant challenges even with relatively small network sizes. Recent studies exist in modeling WMN routing as a mathematical programming problem [12]–[14]. Classic sub-gradient search methods and convex duality theory are then applied [13]. Dynamic links have also been considered, and Lyapunov stability theory is used to solve the problem [14]. These solutions require knowledge of the global network topology and the traffic matrix. Our model instead targets distributed node operations, and focuses on the intrinsic uncertainty of the wireless links. Rather than collecting the traffic matrix in advance, we integrate realtime channel probing and historical statistical information to achieve delay-minimized routing, with online next-hop selection.

This work also differs form on-demand routing protocols such as DSR [3], DSDV [5], and AODV [4]. These protocols are designed mostly to cope with node movement and longer time scale topology changes in wireless networks. Recent studies have also been conducted on examining different routing metrics in WMNs, including expected transmission count in ETX [6], expected transmission time as a function of loss and link bandwidth [15], and total medium access time [16]. Given the stationary nature of mesh routers, these protocols are generally rendered to a semi-deterministic routing protocol. Our protocol however retains dynamic neighbor selection for every packet transmission, so as to fully explore the potentials of multiple neighbors. This implicitly enables multi-paths, yet departs from previous multi-path designs that rely on pre-calculation for dedicated secondary routes [7], [8].

ExOR [9] pioneered research of opportunistic routing (OR) in wireless networks, and made a practical proposal of using a forwarding set to replace a single next-hop candidate. Recently, OR combined with rate adaptation and multi-rate transmission were also examined [10], [17]. Contrasting the OR algorithms, we perform next-hop selection among the candidate set *before*, instead of *after*, the current node transmission. As a result, we avoid the distributed coordination phase required for achieving consensus on a single forwarder. Our multi-rate routing algorithm is of an on-line fashion, and exploits real-time available rate information that can be obtained using an independent rate adaptation module, such as one based on the signal-to-noise-ratio (SNR).

Our work is inspired in part by the recent advances in stochastic routing and stopping theory [1], [11], [18]. Existing formulations of stochastic routing however are usually NP-hard and are not customized for wireless mesh networks. The stopping theory has been recently applied to MAC layer design, particularly in extending the DCF mechanism [19]–[21]. We for the first time demonstrate its application in WMNs routing.

III. Model and Motivation

In this section, we first describe the network and transmission model, and then present a simple example that motivates our study.

A. Wireless Mesh Network and Transmission Model

A WMN can be modeled as a directed graph $G(V, E)$, where V is the set of mesh routers and E is the set of possible wireless links. A unicast route is from a source $n_s \in V$ to a destination $n_d \in V$, connected through multiple relay routers.

A wireless link (n_i, n_j) is generally unreliable due to noise and interference, and is associated with a failure probability p_{n_i,n_j} for not able to start a transmission at a given time point. We first start from a *two-state* model: at any specific time, each link is either *failed* with zero transmission rate, or *working* with full transmission rate r_{n_i,n_j} (with probability $q_{n_i,n_j} = 1 - p_{n_i,n_j}$).

In this binary link model, when a node n_i has a packet for a neighbor n_j, it needs to probe the link. Such a probe can be conducted through a number of techniques, *e.g.*, using the 802.11 RTS/CTS control messages [20], [22]. If the link is working, a transmission can be initialized; otherwise, node n_i waits for time $\Delta\mathbb{T}$, which includes the time to wait for neighboring transmissions that cause interference and the random back-off time in the 802.11 MAC. This link transmission process is illustrated in Fig. 1, where τ is the mean value of $\Delta\mathbb{T}$.

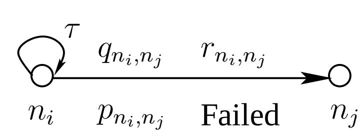

Fig. 1. Link and transmission model.

The expected delay from n_i to n_j can be derived as

$$\mathrm{E}(t) = (qB/r + p\tau)/q \ , \tag{1}$$

where B is the packet size.

This generic model is widely adopted in the literature of wireless networks, for capturing the fundamental uncertainty of wireless transmissions. A number of routing protocols were designed using this model with diverse routing metrics [3]–[6]. Unfortunately, they are in general *deterministic* in the sense that these routing algorithms strike to discover and maintain a static route between periodical route updates. We next use an example to illustrate that this approach does not fully explore the potentials of WMNs.

B. A Motivating Example for Stochastic Routing

Fig. 2 depicts four nodes n_s, n_1, n_2, n_d, connected by four wireless links, with failure probabilities labelled.

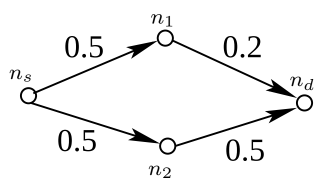

Fig. 2. An example WMN session from n_s to n_d, with link failure probability beside each link.

For a routing request from source n_s to destination n_d, traditional routing algorithms will choose n_1 as the relay, because on average link (n_1, n_d) is better than (n_2, n_d). Assume $B/r = 1$ and $\tau = 1$, we use Eq. (1) to calculate the expected delay from n_s to n_d of the two possible routes:

$$n_s \rightarrow n_1 \rightarrow n_d : \frac{0.5 * 1 + 0.5 * 1}{0.5} + \frac{0.8 * 1 + 0.2 * 1}{0.8} = 3.25$$
$$n_s \rightarrow n_2 \rightarrow n_d : \frac{0.5 * 1 + 0.5 * 1}{0.5} + \frac{0.5 * 1 + 0.5 * 1}{0.5} = 4$$

Therefore the choice of traditional deterministic routing seems rational. Unfortunately, this lower delay of 3.25 is actually far from the optimal, at 2.83. We will give rigorous derivation of the optimal delay later in Section IV. For now, an intuitive explanation is that, although on average better than (n_2, n_d), link (n_1, n_d) remains unreliable and fails for 20% of the time. If we stick to this link, we will suffer from such failures from time to time, even the other link (n_2, n_d)) may be working in the mean time.

This observation motivates our design of stochastic online routing algorithms under link uncertainty. We suggest that each node dynamically determines its next hop through online link status probes; only when all the available links are failed will the node have to back off. Different than traditional multi-path routing, there is no pre-computed next-hop neighbor set. Different than opportunistic routing, a calculated next hop selection is performed *before*, not *after*, the real packet transmission.

IV. Stochastic Routing with Link Uncertainty

While the stochastic routing approach appears promising, there are a number of challenges in realizing it for WMNs. In this section, we first introduce the Stochastic Recoverable Canadian Traveler Problem (SRCTP) [18]. We show the underlying connections between SRCTP and WMN routing, identify and overcome the difficulties in mapping the latter to the former, and develop an online delay-optimal routing protocol under link uncertainty.

A. The Canadian Traveler Problem

The Canadian Traveler Problem (CTP), first introduced by Papadimitriou and Yannakakis in 1989, is an online optimization problem with incomplete information [1]. Consider a traveler in Canada who wants to drive from Vancouver to Toronto in winter. The traveler has a map with road directions and distances. However, given the severe weather conditions, some roads may be blocked by snowfalls, and such blockage would only be revealed when the traveler reaches an adjacent city. The problem is to devise a travel strategy with minimum expected travel time.

Devising a strategy that grantees a constant competitive ratio for CTP is PSPACE-Complete [18]. Approximate solutions have been applied in fields such as transportation, planning, robot navigation [23]. With wireless link failures, naturally, the WMN routing problem also resembles CTP. However, we do have extra information on link statistics, *e.g.*, failure probabilities, which are not available in original CTP. A CTP variant with link blockage probabilities, Stochastic Recoverable-CTP (SRCTP), was examined in [18]. Interestingly, the extra information can indeed lead to an optimal polynomial-time solution: at each node, the next hop selection can be based on a priority list of the neighbors; the list is calculated with a shortest-path-like algorithm, and the first non-blocked neighbor from the list will be chosen.

Our study of stochastic WMN routing is motivated by SRCTP, which admits an optimal online strategy for minimizing expected traveling time. Nonetheless, the mapping from WMN routing to SRCTP remains non-trivial. Neither CTP or SRCTP considers the cost to discover non-blocked roads, which plays an important role in our design. We next address this challenge through a judicious probing strategy.

B. SRCTP-based Routing

Our online SRCTP-based routing protocol relies on a simple local operation, as illustrated in Fig. 3. Here, node n_i has k neighbor nodes $\{n_i^1, \ldots, n_i^k\}$, among which h neighbors are chosen as a *candidate set* for the next hop. Different from conventional routing protocols, we do not fix a neighbor as next hop based on average or historical measures, but dynamically select one through a probing process.

Specifically, n_i will first probe link (n_i, n_i^1), and will use that link if it works. If it fails, the next neighbor link (n_i, n_i^2) will be probed, and so forth until a link works or all the candidates have been probed. In the latter, n_i will back off for a random time with mean value τ. This back off time should be comparable to a packet transmission time to avoid collision. In this case, it also mitigates the dependency of link failures. Note that although a broadcast probe alternative is possible, it requires both a distributed response coordination and modifications to the existing 802.11 protocol, and is therefore undesirable. After the back off, a *round* finishes and n_i restarts another round of probing, and continues until finding a working link to transmit the data.

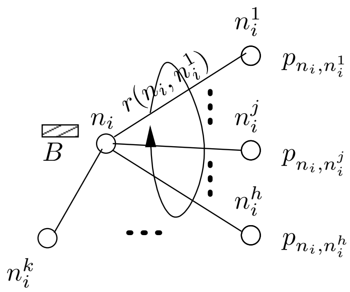

Fig. 3. Round robin probing of next hop candidates.

The delay performance of our protocol closely depends on the probing order of the neighbors. Like in conventional routing protocols, each node n_j keeps a list of the expected delay $\mathrm{E}(t(n_j, n_d))$ from itself to each destinations n_d, and such a list is exchanged between neighbors. Then, for node n_i, the expected delay of using a particular neighbor includes the link probing cost, the transmission time using this link, and the expected delay from the next hop node to the destination. For example, assume 802.11 RTS/CTS handshaking is used for channel probing. If the size of channel probing packets during handshaking is b, and the total inter-frame space included is T_{IFS}, then the cost for a channel probe can be calculated as:

$$T_{probe}(n_i, n_i^j) = 2b/r_{n_i,n_i^j} + T_{IFS} \tag{2}$$

The expected delay for a working link can then be formulated as

$$I_j \triangleq T_{probe}(n_i, n_i^j) + B/r_{n_i,n_i^j} + \mathrm{E}(t(n_i^j, n_d)) \tag{3}$$

The smaller I is, the shorter delay can be expected. The neighbors should therefore be sorted in ascending order of their I values.

C. Size of Candidate Set

The size of the candidate set h also plays an important role in delay minimization. Intuitively, increasing h offers a better chance of finding a working link; yet it increases the probing cost, and a neighbor link with very high I value, even if working, would be of less interest for packet delivery. In that case, it is better to start another round of probing, and hopefully find a 'better' working neighbor.

To derive the optimal h, we insert adjacent links into the candidate set one-by-one, in ascending order of their I values. After each insertion, we update the expected delay to n_d using the temporary candidate set. The insertion continues until the expected delay increases. Let $\mathcal{C}_{n_i}$ be the candidate set so far. Similar to Eq. (1), we can obtain the expected delay as [1]

$$\mathrm{E}_1(t(n_i, n_d)) = (q_1 I_1 + p_1 \tau)/q_1 \tag{4}$$

[1]Since here we focus only on node n_i, we use q_j to represent q_{n_i,n_i^j} and p_i to represent p_{n_i,n_i^j} for ease of exposition.

For h neighbors in $\mathcal{C}_{n_i}$, let

$$P_h \triangleq \prod_{j=1}^{h} p_j, \qquad \alpha_h \triangleq \sum_{j=1}^{h} P_{j-1} q_j I_j \tag{5}$$

According to the probing process, the expected delay from n_i to n_d can be calculated as

$$\mathrm{E}_h(t(n_i, n_d)) = (\alpha_h + P_h \tau)/(1 - P_h) \tag{6}$$

Theorem 1: The trend (increasing or decreasing) of $\mathrm{E}_h(t(n_i, n_d))$ to h solely depends on the sign of δ_h:

$$\delta_h \triangleq (1 - P_h) I_{h+1} - (\alpha_h + \tau) \tag{7}$$

Proof: See Appendix A for details. ■

If $\delta_h < 0$, the expected delay can be reduced by adding the $(h+1)$-th neighbor into the candidate set $\mathcal{C}_{n_i}$. If $\delta_h \geqslant 0$, there is no benefit of doing so.

Lemma 1: For $\forall h'$, if $\mathrm{E}_{h'+1}(n_i, n_d) \geqslant \mathrm{E}_{h'}(n_i, n_d)$, then $\forall h > h', \mathrm{E}_{h+1}(n_i, n_d) \geqslant \mathrm{E}_h(n_i, n_d)$.

Proof: Since $\mathrm{E}_{h'+1}(n_i, n_d) \geqslant \mathrm{E}_{h'}(n_i, n_d)$, we have $\delta_{h'} \geqslant 0$ according to Theorem 1.

Since the neighbor links are sorted in ascending order of I in Eq. (3), we have $I_{h'+2} \geqslant I_{h'+1}$. It follows that $\delta_{h'+1} \geqslant 0$ (See Appendix B). According to Theorem 1, the lemma is true. ■

For the optimal size of candidate set, we try to find the smallest h' (denoted as h^*) that satisfies $\mathrm{E}_{h^*+1}(n_i, n_d) \geqslant \mathrm{E}_{h^*}(n_i, n_d)$. If such h^* exists, it will be set to the optimal size of the candidate set. If there does not exist such h^*, *i.e.*, $\mathrm{E}_k(n_i, n_d) < \mathrm{E}_{k-1}(n_i, n_d)$, the candidate set will include all the k neighbors.

Lemma 1 also confirms the rationality of the aforementioned neighbor probing order. From the proof, we can see that h^* can simply be calculated through checking the sign of δ_h in Eq. (7). We add more neighbor nodes into $\mathcal{C}_{n_i}$, until $\delta_h \geqslant 0$ or $h = k$. Altogether, we have the following theorem.

Theorem 2: SRCTP routing algorithm minimizes the expected delivering delay under the two-state link model.

The rest of the proof to the theorem, besides Lemma 1, is similar to that of the stochastic Recoverable-CTP algorithm, and is omitted here due to space limitation.

V. Stochastic Delay Optimization with Multi-Value Link Capacities

In real-world scenarios, depending on the status of the physical medium, the level of noise and interference, as well as the modulation scheme chosen, a link may operate at a number of different transmission rates between zero and the full capacity. For example, an 802.11b link can operate at 1Mbps, 2Mbps, 5.5Mbps or 11Mbps. Stochastic delay optimization becomes more complex in the multi-value link capacity model. In particular, the SRCTP algorithm can not be easily extended, since it has no mechanism of comparatively evaluating the option of transmitting through a favorable link at an intermediate capacity. We introduce techniques from stopping theory [11], and combine them with online neighbor selection from Sec. IV, for computing new routing strategies that target optimal expected routing delay with the presence of dynamic multi-rate links.

A. The Multi-rate Link Model and Optimal Stopping

Assume each link in $G(V, E)$ has a set of discrete, limited available transmission rates $R_{n_i,n_i^j} = \{r_0, r_1, \dots, r_{m_j}\}$, where $r_0 = 0$ and $r_k < r_{k+1}$, $\forall 0 \le k < m_j$, *i.e.*, R_{n_i,n_i^j} is sorted in ascending order. Realtime link rate is a random variable $\mathbb{R}_{n_i,n_i^j}$. The probability mass function of $\mathbb{R}_{n_i,n_i^j}$ is

$$f_{\mathbb{R}}(r_{k_j}) = \Pr(\mathbb{R} = r_{k_j}) = p_{k_j}, \qquad 1 \leqslant k_j \leqslant m_j \tag{8}$$

A new challenge in the multi-rate link model is that at a given time, favorable links may have an intermediate transmission rate that is neither zero nor the full capacity r_{m_j}. It is non-trivial to determine which option eventually leads to a better expected delay: (a) starting the transmission now, or (b) wait and probe more neighbors, and probably wait till the next round. Techniques in stopping theory are designed for decision making with uncertain future, and are naturally suited for our delay optimization problem here. We next briefly introduce related concepts from stopping theory and design a thresh-hold value based routing solution.

In stopping theory, a *stopping rule problem* is defined by two objects: (i) a sequence of random variables $\mathbb{X}_1, \dots, \mathbb{X}_n$ with known joint distribution, and (ii) a sequence of real-valued reward functions $y_0, y_1(x_1), y_2(x_1, x_2), \dots$. A player can make as many observations $\mathbb{X}_1 = x_1, \dots, \mathbb{X}_k = x_k$ as she wishes before stopping at round k, and receives the known *reward* $y_k(x_1, \dots, x_k)$. The goal is to optimize the expected reward $E\{y_k\}$.

In a more specific version, the *Markov stopping problem*, the reward y_k depends only on the latest observation x_k, and $\mathbb{X}_1, \mathbb{X}_2, \dots$ follow independent identical distributions. The problem of optimally stopping probing and starting transmission in WMN routing can be modeled as a Markov stopping problem as follows.

We define the observation as the expected minimum delay after each round of channel probing. Assume h neighbors are selected into the candidate set $\mathcal{C}_{n_i}$. Then such minimal expected delay, $\mathbb{X}(h)$, can be computed as

$$\mathbb{X}(h) \triangleq T_{probe}(h) + \min_{1 \leqslant j \leqslant h} \left(\frac{B}{\mathbb{R}_{n_i,n_i^j}} + \mathrm{E}(t(n_i^j, n_d))\right) \tag{9}$$

in which

$$T_{probe}(h) \triangleq \sum_{j=1}^{h} T_{probe}(n_i, n_i^j) \tag{10}$$

Next, we can construct the reward sequence $\mathbb{Y}_n(h)$ based on the stochastic routing decision process as

$$\mathbb{Y}_n(h) \triangleq \mathbb{X}(h) + (n-1)\tau \tag{11}$$

An optimal stopping rule exists if the sequence above satisfy the following two conditions ([11] Ch3):

$$A1.\ \mathrm{E}\{\inf_n \mathbb{Y}_n\} > -\infty$$
$$A2.\ \liminf_{n\to\infty} \mathbb{Y}_n \geqslant \mathbb{Y}_\infty \text{ almost surely}$$

It is easy to verify that $\mathbb{Y}_n(h)$ defined in (11) satisfies both $A1$ and $A2$ (see Appendix C). The existence of an optimal stopping rule is consequently guaranteed. The rule is based on a computed threshold value $V^*(h)$, and states that we should stop at the $N^*(h)$-th round where the expected delay is below $V^*(h)$, that is

$$N^*(h) = \min\{n \geqslant 1 : \mathbb{X}(h) \leqslant V^*(h)\} \tag{12}$$

Once the threshold value $V^*(h)$ is reached, it's time to stop probing and start transmission through link n_{ij^*}, with

$$j^* = \arg\min_{1 \leqslant j \leqslant h} \left(\frac{B}{r_j(N^*(h))} + \mathrm{E}(t(n_i^j, n_d))\right) \tag{13}$$

where $r_j(N^*(h))$ is the real-time available transmission rate from n_i to n_i^j at the $N^*(h)$-th round.

We are now ready to design *the ST routing algorithm*: the stochastic routing algorithm for multi-rate links based on the optimal stopping rule.

B. The ST Routing Algorithm

Our ST routing algorithm relies on the local routing decision process, a Markov stopping problem whose best strategy is described in the optimal stopping rule in (12). To support such decision making, we need to calculate two key parameters. One is the optimal threshold $V^*(h)$, for stopping probing and starting transmission. The other is the optimal candidate set $\mathcal{C}^*$ with size h^*. Below we describe how $V^*(h)$, the delay table, and $\mathcal{C}^*$ can be computed.

Computing $V^*(h)$
By Markov stopping theory, the optimal threshold value $V^*(h)$ can be found through solving the following equation:

$$V^*(h) = \mathrm{E}\min\{\mathbb{X}(h), V^*(h)\} + \tau \tag{14}$$

From the multi-rate link model in (8) and the definition of $\mathbb{X}(h)$ in (9), we know $\mathbb{X}(h)$ is a discrete random variable. Consequently, $V^*(h)$ must take a value from the domain of $\mathbb{X}(h)$: $S_{x,h} = \{x_1, \dots, x_{w(h)}\}$, which is sorted in ascending order and has corresponding probabilities $\{\hat{p}_1, \dots, \hat{p}_{w(h)}\}$, with $\sum_{u=1}^{w(h)} \hat{p}_u = 1$. Let x_u be the value taken by $V^*(h)$, with some $1 \leqslant u \leqslant w(h)$. Equation (14) can then be translated into

$$x_u = \sum_{i=1}^{u-1} x_i \hat{p}_i + \sum_{i=u}^{w(h)} x_u \hat{p}_i + \tau \tag{15}$$

which can be further manipulated into

$$\sum_{i=1}^{u-1} (x_u - x_i)\hat{p}_i = \tau \tag{16}$$

In practice, the following equation can be used to compute a value for x_u:

$$x_u = max\{x_w | \sum_{i=1}^{w-1} (x_w - x_i)\hat{p}_i \leqslant \tau\} \tag{17}$$

Knowing the probability mass functions of the h links, according to the definition of $\mathbb{X}(h)$, we can calculate elements in $S_{x,h}$ as

$$x_{k,j} = T_{probe}(h) + \frac{B}{r_{k_j}} + \mathrm{E}(t(n_i^j, n_d)), 1 \leqslant k \leqslant m_j \tag{18}$$

There are at most $\sum_{j=1}^{h} m_j + 1$ possible values, including one for the case where all h links fail. So, $w(h) = \sum_{j=1}^{h} m_j + 1$. And $\{k, j\}$ means the k-th available rate to the j-th neighbor, which corresponds to one unique index between 1 and $w(h)$. Next, $\hat{p}_v$ can be calculated as follows, assuming x_v is obtained at link (n_i, n_{ij_v}) at the rate of $r_{k_{v,j}}$:

$$\hat{p}_v = p_{k_{v,j}} \prod_{j=1, j\neq j_v}^{h} \sum_{k=k_{v,j}}^{m_j} p_{k_j} \tag{19}$$

in which $k_{v,j}$ corresponding to the index of $r_{k_{v,j}}$ with its $x_{k,j}$ value calculated by (18) satisfies $x_{k,j} > x_v$.

Now we can finally solve the optimal $V^*(h)$ as x_u from (16). Then, the optimal stopping rule in (12) defines the routing decision process. Channel probing will proceed to the next round if $x_n > V^*(h)$, till we get the first $x_{N^*} \leqslant V^*(h)$.

Building the Delay Tables

The probability of $\mathbb{X}(h) \leqslant V^*(h)$ can be calculated as

$$\dot{q} \triangleq \Pr(\mathbb{X}(h) \leqslant V^*(h)) = \sum_{i=1}^{u} \hat{p}_i \tag{20}$$

And the expected $\mathbb{X}_h$ value in the precondition of $\mathbb{X}(h) \leqslant V^*(h)$ can be calculated as

$$\mathrm{E}(\mathbb{X}(h) | \mathbb{X}(h) \leqslant V^*(h)) = \frac{1}{\dot{q}} \sum_{i=1}^{u} x_i \hat{p}_i \tag{21}$$

Let $\dot{p} \triangleq 1 - \dot{q}$, we can compute the optimal expected delay from n_i to n_d for the delay table, required for computing $V^*(h)$, as following:

$$\mathrm{E}_h(t(n_i, n_d)) = (\dot{q}\mathrm{E}(\mathbb{X}(h)|\mathbb{X}(h) \leqslant V^*(h)) + \dot{p}\tau)/\dot{q} \tag{22}$$

Computing $\mathcal{C}_{n_i}$

We next describe how to compute the optimal h and candidate set $\mathcal{C}_{n_i}$. Intuitively, increasing h incurs extra channel probing cost, which will increase $\mathbb{X}(h)$ and therefore cause V^* and $\mathrm{E}_h(t(n_i, n_d))$ to increase. However, if the minimal expected delay $x_{m_{h+1}, h+1}$ through the $n_{i,h+1}$ neighbor is smaller than $V^*(h)$, it is still possible to have $V^*(h+1) < V^*(h)$, so as $\mathrm{E}_{h+1}(t(n_i, n_d)) < \mathrm{E}_h(t(n_i, n_d))$.

Inspired by such an intuition, we compute h and $\mathcal{C}_{n_i}$ through the following. First, assume n_i has k neighbors $\{n_i^1, \ldots, n_i^k\}$, computer for each $1 \leqslant j \leqslant k$:

$$\hat{I}_j = T_{probe}(n_i, n_i^j) + \frac{B}{r_{m_j}} + \mathrm{E}(t(n_i^j, n_d)) \tag{23}$$

The algorithm of computing the candidate set $\mathcal{C}_{n_i}$ is then shown in Algorithm 1.

Algorithm 1 Computing Candidate Set $\mathcal{C}_{n_i}$

Require: n_i, n_d,
  $\mathcal{N}_{n_i} = \{n_i^1, \ldots, n_i^k\}$, *# neighbor node set sorted by* $\hat{I}_j$
  $\hat{I} = \{\hat{I}_1, \ldots, \hat{I}_k\}$, *# sorted in ascending order*
  $\mathrm{E}(t(n_i^j, n_d))$, $1 \leqslant j \leqslant k$,

```
C_{n_i} ⇐ {n_i^1}, h ⇐ 1
Calculate T_probe(h) by (10), V*(h) by (14), E_h(t(n_i, n_d)) by (22)
N_{n_i} ⇐ N_{n_i} / {n_i^1}
while N_{n_i} ≠ ∅ do
   n_curr ⇐ NULL
   E_min ⇐ E_h(t(n_i, n_d))
   for each n_i^j in N_{n_i} do
      if T_probe(h) + Î_j > V*(h) then
         N_{n_i} ⇐ N_{n_i} / {n_i^j}  # reject 'bad' neighbor
      else
         C_{n_i} ⇐ C_{n_i} ⋃ {n_i^j}  # temporary insertion to C_{n_i}
         Calculate E_{h+1}(t(n_i, n_d)) by (22)
         if E_{h+1}(t(n_i, n_d)) < E_min then
            n_curr ⇐ n_i^j
            E_min ⇐ E_{h+1}(t(n_i, n_d))
         end if
         C_{n_i} ⇐ C_{n_i} / {n_i^j}
      end if
   end for
   if n_curr == NULL then
      break
   end if
   C_{n_i} ⇐ C_{n_i} ⋃ {n_curr}
   N_{n_i} ⇐ N_{n_i} / {n_curr}
   h ⇐ h + 1
   E_h(t(n_i, n_d)) ⇐ E_min
   Update T_probe(h) and V*(h)
end while
```

Algorithm 1 expands $\mathcal{C}_{n_i}$ by one neighbor at a time. A neighbor is included only if its presence leads to lower expected delay, *i.e.*, $T_{probe}(h) + \hat{I}_j < V^*(h)$. The complexity of Algorithm 1 is $O(k^2)$. In comparison, a naive algorithm that examines all possible combinations of neighbors has a complexity of $O(\sum_{u=1}^{k} C(k, u)) = O(2^k - 1) = O(2^k)$.

With optimal h^* and $\mathcal{C}_{n_i}^*$, using the optimal stopping rule $N^*(h^*)$ in (12), we can guarantee the optimal expected delay from n_i to n_d, *i.e.*, the ST routing decision is optimal.

C. Connection between the SRCTP and ST Algorithms

We now take a retrospect at the SRCTP and ST algorithms presented for the two-state and multi-state link models respectively, and show an interesting connection between them:

although the two algorithms have been designed from different perspectives and using different techniques, ST indeed degrades into SRCTP when applied to WMNs with two-state links only. This is in line with the respective optimality of the two algorithms, and will be further confirmed through simulation studies in Sec. VI.

With binary links (zero or full capacity), if we omit the link probing cost $T_{probe}(h)$ defined in (10), the order in $S_{x,h}$ is exactly the same as the order of neighbor links in SRCTP, which uses (3) as the sorting criteria. Furthermore, the equation for computing $\hat{p}_v$ in (19) degenerates into $q_v P_{v-1}$, in which P_v is defined in (5). The optimal threshold equation given in (14), extended to (15) under discrete probability distribution, is exactly the equation $\delta_h = 0$ defined in (7). Next, $\delta_h = 0$ is the optimal boundary for $\mathrm{E}_h(t(n_i, n_d))$, which takes the same role as the optimal equation (14) in the ST routing algorithm. In conclusion, the SRCTP and the ST algorithms unify under the two-state link model, achieving the same optimal end-to-end routing delay.

VI. Performance Evaluation

We now present simulation studies of the proposed SRCTP and ST routing algorithms, and compare them to wireless routing algorithms with fixed neighbor selection. We have implemented SRCTP, ST, and traditional semi-deterministic routing using a dedicated simulator written in C++, with ns-2 generated traffic trace and network topologies. We use the dei9011 mr library [24], [25] that accompanies the ns-2 allinone package for producing multi-rate MAC-layer links, based on SNR rate adaption. A revised version of dei9011 mr is utilized for simulating two-state links by depressing transmission rates that are lower than maximum, *i.e.*, 11Mbps for 802.11b and 54Mbps for 802.11g.

We evaluate SRCTP and ST on both grid and random WMN topologies. The first is a 5×5 grid network with neighbor distance of 100m, same as in previous studies [26]. Transmission power is adjusted so that node transmission range is close to 150m. The SNR rate adaptor is used and no explicit transmission range setting is required. For random topologys, we used the Setdest tool from the ns-2 allinone package. We deploy 100 nodes in a $1200m\times1200m$ area [27].

We compare SRCTP and ST with traditional routing such as ETX [6], DSR [3] and AODV [4]. For the latter, we select the first node in the candidate set built by our algorithms, and compare the performance against SRCTP and ST in terms of end-to-end routing delay and packet delivery ratio.

A. *SRCTP* vs. *Semi-deterministic Routing: Delay*

Grid Network Topology

Fig. 4 shows end-to-end delay of SRCTP and of deterministic routing, in the grid network topology, with 300 packets routed from the bottom-left node to the top-right node. Only 50 packets are shown, for figure clarity; the trend for 300 packets is similar. Fig. 5 depicts the same delays in ascending order. First, we observe that SRCTP outperforms deterministic routing in most except a few cases, by providing lower delays.

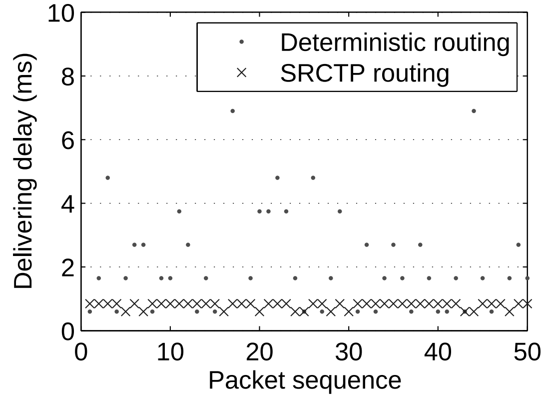

Fig. 4. End-to-end delay in grid network, SRCTP *vs.* deterministic routing.

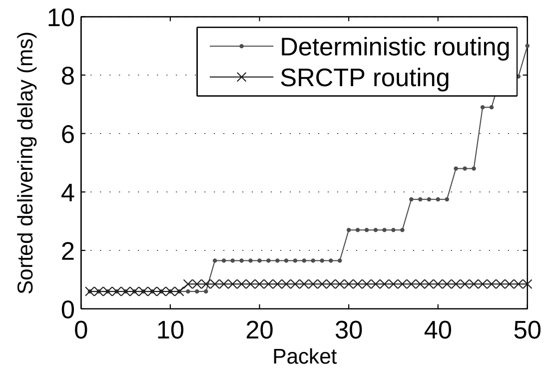

Fig. 5. Sorted delays in grid network, SRCTP *vs.* deterministic routing.

The greedy nature of deterministic routing ignores possible failures of the most promising link that can happen at realtime. SRCTP instead performs online optimization and makes more judicious routing decisions among a larger set of candidate neighbors. Second, packet delays using SRCTP is much more stable than those using deterministic routing. Although individual links may have drastically different capacities at different times, SRCTP can avoid 'bad' links at realtime and the overall delay along the entire path is relatively stable. In practice, that will in turn translate into better in-oder packet arrivals at the receiver side and higher TCP throughput.

Random Network Topology

Fig. 6 depicts the sorted packet delivering delays in the random network topology. Since the network is larger and the topology less regular, the packet delays are less stable for both SRCTP and deterministic routing. Nonetheless, SRCTP still demonstrates relatively better stability, with a highest delay of 15 ms, smaller than $20+$ ms of deterministic routing. Furthermore, as in the grid topology, SRCTP leads to smaller end-to-end delay for most of the packets measured.

B. *SRCTP* vs. *Semi-deterministic Routing: Loss Ratio*

We conducted further simulations for measuring and comparing the packet loss ratio. Each packet is attempted for a maximum of 10 times for transmission at a given node, and dropped after that. Fig. 7 depicts drop ratios of both SRCTP and deterministic routing. Unicast sessions, each containing

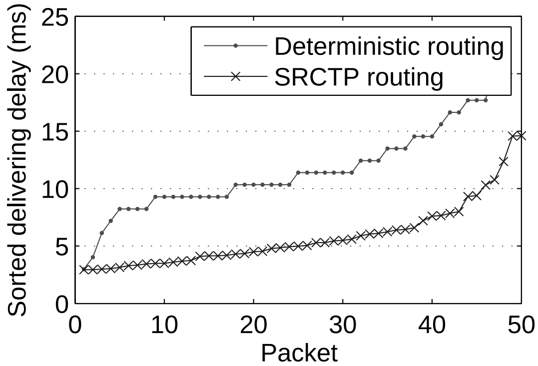

Fig. 6. Sorted packet delays in the random topology.

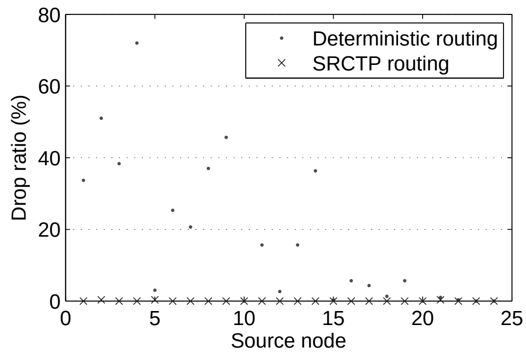

Fig. 7. Packet drop ratios in the grid network.

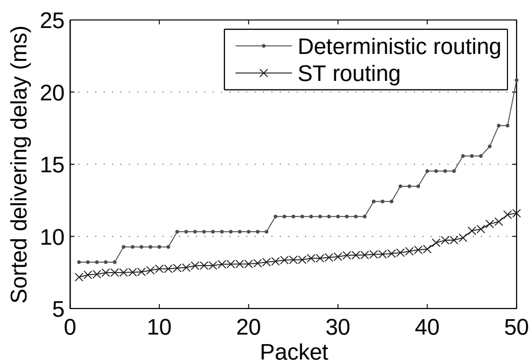

Fig. 8. Sorted delays in the random topology. ST *vs.* deterministic routing.

300 packets, are initialed from all other 24 nodes to the destination. Here the advantage of SRCTP is rather evident. A well-designed deterministic routing algorithm, such as DSR or AODV, would re-establish a new route upon a link failure, and the loss rate in practice will hence be more moderate than observed here. Nonetheless, we believe that SRCTP should still be superior since making online routing decisions at the link level is more flexible and effective than constantly re-computing the entire routing path.

C. ST vs. *Semi-deterministic Routing*

Fig. 8 shows the simulation results of ST routing algorithm compared with deterministic routing, with multi-rate wireless links, in the random network topology. We can observe that end-to-end packet delays show higher stability than in the previous two-state case. Overall, the ST algorithm outperforms

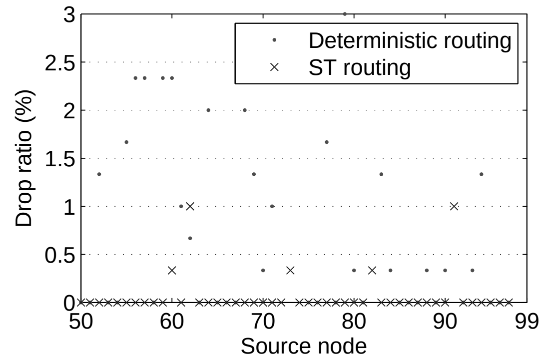

Fig. 9. Packet drop ratios in the random topology, ST *vs* deterministic routing.

deterministic routing by providing lower average packet delays and higher delay stability. Again, this can be contributed to the dynamic neighbor selection in ST routing.

Fig. 9. depicts the packet drop ratio comparison, with 50 unicast sessions established. ST suffers much lower drop ratios than deterministic routing does. The figure suggests that packet drops happen rather randomly. In most cases, ST routing can avoid temporary link failures and result in near-zero drop ratios. However, there do exist a few cases when prolonged ‘bad’ conditions happen across the entire candidate set, leading to eventual packet drops. The frequency of such event is much lower than a single link failure though, which is sufficient to cause a packet drop in semi-deterministic routing.

D. SRCTP vs. *ST: Two-State Link Model*

Fig. 10 depicts the comparison between SRCTP and ST, with two-state wireless links, in a randomly generated network topology. Simulations are conducted both with and without considering link probing cost. Without link probing cost, SRCTP and ST lead to rather close delays, confirming our discussions on unified optimality in Section V-C. With link probing cost, ST performs slightly worse than SRCTP. One factor that contributes to this difference is that the ST always probes all adjacent links in the candidate set, and the routing table establishment algorithm also takes this effect into consideration for computing optimal routing.

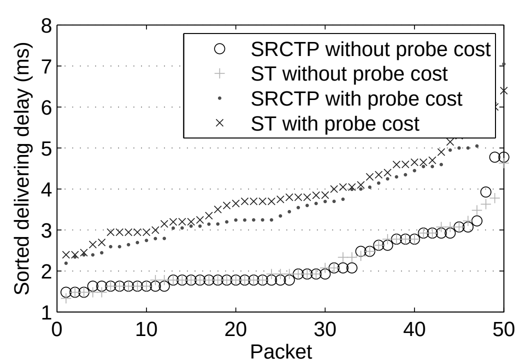

Fig. 10. Sorted packet delays, random topology, SRCTP *vs.* ST.

VII. CONCLUSIONS

Wireless links are inherently stochastic and uncertain in that their quality and effective transmission rates are volatile, varying at both long and short time scales. This work proposes to optimize end-to-end packet routing delay with explicit consideration of such uncertainty in link qualities. We depart from the prefixed deterministic route selection, and apply online, stochastic route optimization instead. We model minimum delay routing as a stochastic optimization problem, and draw inspiration from the classic Canadian Traveler Problem as well as from the Optimal Stopping Problem, to design routing solutions for two-state links and multi-rate links, respectively. Analysis and simulation results both show that the two solutions naturally unify in the two-state link model. Simulation results further demonstrate considerable improvement over traditionally deterministic packet routing, in both end-to-end delay and packet delivery ratio, suggesting that the new online routing philosophy is indeed better suited for the nature of WMNs.

REFERENCES

[1] C. H. Papadimitriou and M. Yannakakis, "Shortest paths without a map," in *Proc. of ICALP*. London, UK: Springer-Verlag, 1989, pp. 610–620.

[2] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47(4), pp. 445–487, 2005.

[3] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," *Kluwer International Series in Engineering and Computer Science*, pp. 153–179, 1996.

[4] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. of Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.

[5] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, p. 244, 1994.

[6] D. Couto, D. S. J., D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networking*, vol. 11, no. 4, pp. 419–434, 2005.

[7] N. S. Nandiraju, D. S. Nandiraju, and D. P. Agrawal, "Multipath routing in wireless mesh networks," in *Proc. of IEEE MASS*, 2006, pp. 741–746.

[8] W.-H. Tarn and Y.-C. Tseng, "Joint multi-channel link layer and multi-path routing design for wireless mesh networks," in *Proc. of IEEE INFOCOM*, 2007, pp. 2081–2089.

[9] S. Biswas and R. Morris, "Exor: opportunistic multi-hop routing for wireless networks," in *Proc. ACM SIGCOMM*. New York, NY, USA: ACM, 2005, pp. 133–144.

[10] R. Laufer, H. Dubois-Ferriere, and L. Kleinrock, "Multirate anypath routing in wireless mesh networks," in *Proc. IEEE INFOCOM*, 2009, pp. 37–45.

[11] T. Ferguson, *Optimal Stopping and Applications*, Mathematics Department, UCLA, 2006. [Online]. Available: www.math.ucla.edu/~tom/Stopping/Contents.html

[12] R. L. Cruz and A. V. Santhanam, "Optimal routing, link scheduling and power control in multihop wireless networks," in *Proc. of IEEE INFOCOM*, vol. 1, 2003, pp. 702–711.

[13] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, "Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in *Proc. of IEEE INFOCOM*, 2006, pp. 1–13.

[14] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1–144, 2006.

[15] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of ACM MobiCom*, 2004.

[16] B. Awerbuch, D. Holmer, and H. Rubens, "The medium time metric: High throughput route selection in multi-rate ad hoc wireless networks," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 253–266, 2006.

[17] K. Zeng, W. Lou, and H. Zhai, "On end-to-end throughput of opportunistic routing in multirate and multihop wireless networks," in *Proc. INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 2008, pp. 816–824.

[18] A. Bar-Noy and B. Schieber, "The canadian traveller problem," in *Proc. SODA*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1991, pp. 261–270.

[19] P. S. Chandrashekhar Thejaswi, J. Zhang, M.-O. Pun, and H. V. Poor, "Distributed opportunistic scheduling with two-level channel probing," in *Proc. of IEEE INFOCOM 2009*, 2009, pp. 1683–1691.

[20] D. Zheng, W. Ge, and J. Zhang, "Distributed opportunistic scheduling for ad-hoc communications: an optimal stopping approach," in *Proc. of ACM MobiHoc*, NY, USA, 2007, pp. 1–10.

[21] D. Zheng, M.-O. Pun, W. Ge, J. Zhang, and H. V. Poor, "Distributed opportunistic scheduling for ad-hoc communications under noisy channel estimation," in *Proc. of IEEE ICC*, 2008, pp. 3715–3719.

[22] N. B. Chang and M. Liu, "Optimal channel probing and transmission scheduling for opportunistic spectrum access," in *Proc. of ACM MobiCom*, NY, USA, 2007, pp. 27–38.

[23] E. Nikolova and D. Karger, "Route planning under uncertainty: The canadian traveller problem," in *Proc. of the Twenty-Third AAAI Conference on Artificial Intelligence*, 2008, pp. 969–974.

[24] Special Interest research Group on NETworking (SIGNET), "dei802.11mr:a new 802.11 implementation for ns-2,". [Online]. Available: http://www.dei.unipd.it/wdyn/?IDsezione=5090

[25] D. Passos and C. Albuquerque, "A joint approach to routing metrics and rate adaptation in wireless mesh networks," *Proc. of IEEE INFOCOM Workshops*, pp. 1–2, 2009.

[26] A. Das, H. Alazemi, R. Vijayakumar, and S. Roy, "Optimization models for fixed channel assignment in wireless mesh networks with multiple radios," in *Proc. of IEEE SECON*, vol. 5. Citeseer, 2005, pp. 463–474.

[27] P. Hu, A. Pirzada, and M. Portmann, "Experimental evaluation of aodv in a hybrid wireless mesh network," in *Proc. of the 5th Workshop on the Internet, Telecommunications and Signal Processing*, 2006.

APPENDIX

A. Proof of Theorem 1 :

$$\begin{aligned}
\frac{\mathrm{E}_{h+1}(t(n_i,n_d))}{\mathrm{E}_h(t(n_i,n_d))} &= \frac{\alpha_{h+1}+P_{h+1}\tau}{1-P_{h+1}}\cdot\frac{1-P_h}{\alpha_h+P_h\tau}\\
&= \frac{\alpha_h+P_hq_{h+1}+P_hp_{h+1}\tau}{1-P_{h+1}}\cdot\frac{1-P_h}{\alpha_h+P_h\tau}
\end{aligned}$$

$$= \frac{\alpha_h+P_h\tau+P_hq_{h+1}I_{h+1}-P_hq_{h+1}\tau}{1-P_{h+1}}\cdot\frac{1-P_{h+1}-P_hq_{h+1}}{\alpha_h+P_h\tau}$$

$$= 1+\frac{P_hq_{h+1}}{(1+P_{h+1})(\alpha_h+P_h\tau)}\cdot[(1-P_h)I_{h+1}-(\alpha_h+\tau)]$$

$\square$

B. Complementary Proof of Lemma 1 :

$$\begin{aligned}
\delta_{h^*+1} &= (1-P_{h^*+1})I_{h^*+2}-(\alpha_{h^*+1}+\tau)\\
&= (1-P_{h^*+1})I_{h^*+2}-(\alpha_{h^*}+P_{h^*}q_{h^*+1}I_{h^*+1}+\tau)\\
&\geqslant (1-P_{h^*+1})(I_{h^*+2}-I_{h^*+1}) \qquad (\delta_{h^*}\geqslant 0)\\
&\geqslant 0 \qquad (I_{h^*+2}\geqslant I_{h^*+1})
\end{aligned}$$

$\square$

C. Proof of $\mathbb{Y}_n$ satisfy $A1$ and $A2$:

$\mathbb{Y}_n(h)$ in (11) is always greater than 0, therefore $A1$ is satisfied.

Further, since from (11) we know $\mathbb{Y}_n(h)$ is monotonically increasing as n increase, so

$$\begin{aligned}
\liminf_{n\to\infty}\mathbb{Y}_n(h) &= \lim_{n\to\infty}(\inf_{m\geqslant n}\mathbb{Y}_m(h))\\
&= \lim_{n\to\infty}\mathbb{Y}_n(h)\\
&= \mathbb{Y}_\infty(h) \text{ almost surely} \qquad (24)
\end{aligned}$$

Hence $A2$ is also true. $\square$