

A Dynamic Programming Approach for Routing in Wireless Mesh Networks

J. Crichigno[†], J. Khoury[†], M. Y. Wu[‡], W. Shu[†]

[†] Electrical & Computer Engineering Dept., University of New Mexico, Albuquerque - NM, USA

[‡] Dept. of Computer Science & Engineering, Shanghai JiaoTong University, Shanghai, China

Emails: [†]{jcrichigno, jkhoury, wshu}@ece.unm.edu, [‡]wu-my@cs.sjtu.edu.cn

Abstract—The routing problem in Wireless Mesh Networks is concerned with finding “good” source-destination paths. It generally faces multiple objectives to be optimized, such as i) path capacity, which accounts for the bits per second that can be sent along the path connecting the source to the destination node, and ii) end-to-end delay. This paper presents the Mesh Routing Algorithm (MRA), a dynamic programming approach to compute high-capacity paths while simultaneously bounding the end-to-end delay. The proposed algorithm also provides the option of routing through multiple link-disjoint paths, such that the amount of traffic through each path is proportional to its capacity. Simulation results show that MRA outperforms other common techniques in terms of path capacity, while at the same time bounding the end-to-end delay to a desired value.

I. INTRODUCTION

IEEE 802.11-based wireless mesh networks (WMNs) consist of mesh routers and mesh clients equipped with one or more IEEE 802.11 radios, where mesh routers are mainly stationary and provide wireless access to clients. Additionally, mesh routers serve as relay nodes, forwarding packets from other nodes and forming a multi-hop environment [1].

Routing in WMNs is concerned with finding “good” source-destination paths and has been an active research area for several years. Much of the work is focused on the improvement of global resources [2], or on the performance of individual transfers [1], [3], [4]. Either case, the routing problem can be casted as a multi-objective optimization with objectives such as load-balancing, end-to-end delay, and path capacity. The latter refers to the bits per second (bps) that can be sent along the path connecting the source node to the destination node. Load-balancing schemes mitigate hot spots and improve the reliability by routing through multiple paths, while the end-to-end delay and the path capacity are typical Quality of Service (QoS) requirements that need to be optimized or bounded. In general, they can conflict with each other, and meaningful trade-offs have to be made.

To the best of our knowledge, the only technique proposed to simultaneously minimize the end-to-end delay and maximize the path capacity is the weighted sum, whereby an objective function $f = \beta_1 \cdot (\text{end-to-end delay}) + \beta_2 \cdot (\text{path capacity})$ is optimized [1], [3], [4], [5]. Although the idea is simple, the process of appropriately setting β_1 and β_2 can be complex, since i) both objectives take different orders of magnitude; ii) the weights depend on the importance of the relative objective to the problem at hand. For example, different source-destination pairs requiring delay-bounded paths may

impose different delay bounds. Thus, the precise value of the weights capturing QoS requirements can only be performed on a case-by-case basis. Unfortunately, we are not aware of any systematic procedure to determine β_1 and β_2 ; and iii) the weighted sum technique does not guarantee that any optimal trade-off solution can be obtained [6]. Hence, solutions that best satisfy QoS requirements are not guaranteed.

To solve this problem, we propose a routing algorithm called Mesh Routing Algorithm (MRA). MRA is a dynamic programming [7] approach to compute high-capacity, end-to-end delay-bounded paths. Additionally, MRA provides the option of routing through multiple link-disjoint paths, such that the amount of traffic through each path is proportional to its capacity (multi-path option). The proposed algorithm may be useful not only for WMNs, but also for hybrid networks composed of Vehicular Ad hoc Networks (VANETs) and WMNs, where delay-bounded paths are required to deliver safety messages [8]. Recent work has demonstrated that the performance of VANETs can be improved if packets are routed through roadside infrastructure networks [9].

The remainder of this paper is organized as follows. Section II discusses related work. Section III lists our assumptions, and presents the metrics for the end-to-end delay and the path capacity. Section IV describes the proposed routing algorithm. Section V summarizes the simulation results, and Section VI concludes and discusses our future work. We refer the interested reader to [10] for a longer version of this paper.

II. RELATED WORK

So et al. [11] proposed a load-balancing scheme for single-radio networks, which requires consecutive nodes on a path to synchronously switch their interface as needed. When applied to multi-radio networks, however, the scheme does not exploit the fact that nodes can send and receive simultaneously.

De Couto et al. [12] implemented a routing protocol that incorporates a metric called *expected transmission count* (*etx*), which measures the expected number of transmissions to successfully send a packet across a link. The metric reflects the effects of link loss ratios. However, it does not necessarily select good paths when the nominal rate of links can vary¹.

To optimize the end-to-end delay and the path capacity, Draves et al. [1] defined a two-term weighted sum called *weighted cumulative expected transmission time* (*wcett*). The

¹The nominal rate of a link refers to the raw data rate the link is operating at. For example, 802.11b links can operate at 11, 5.5, 2 or 1 Mbps.

first term accounts for the end-to-end delay, while the second term indirectly accounts for the *intraflow*² interference. By minimizing the intraflow interference, the path capacity is increased. Similar weighted sums are proposed in [3], [4]. In [3], the metric attempts to capture *interflow* interference, while in [4] the metric quantifies intraflow interference. The *wcett* metric is also used in [5], where a traffic splitting algorithm to balance the load in tree-structured WMNs is proposed.

Iannone et al. [13] defined the cost of a link as the inverse of its nominal rate, and showed that by finding low-cost paths, the throughput is improved with respect to the hop metric.

III. METRICS FOR DELAY AND PATH CAPACITY

A. Assumptions

We assume that the nodes are stationary and equipped with one or more 802.11 radios. If a node has multiple radios, they are tuned to predetermined non-overlapping channels. The transmission power is fixed, and the RTS/CTS mechanism at the Distributed Coordination Function (DCF) [14] is disabled, since it does not improve performance [15].

B. Metric for end-to-end delay

Let $G = (V, E)$ be the graph representing the WMN, where V is the set of nodes and E the set of links. Let $C = \{c_1, c_2, \dots, c_k\}$ be the set of k orthogonal channels, and $l = (u, v, c)$ be a directed link between nodes u and v , tuned to channel c , such that $l \in E$ and $c \in C$. The DCF [14] specifies that a unicast transmission through a link l may be retransmitted up to 7 times (including the first transmission). Denote $t_i(l)$ as the delay to send a data frame during the i^{th} retransmission, $t_{DATA}(l)$ as the transmission time of the frame (i.e., the time the sending interface accesses the channel), and $CW_i(l)$ as the time spent in backoff state during the i^{th} retransmission. For a data frame of S bytes and a link operating at a nominal rate of $r(l)$ bps, $t_{DATA}(l) = 8S/r(l)$. Note that $t_{DATA}(l)$ and $CW_i(l)$ are associated with the sending interface (i.e., for $l = (u, v, c)$, $t_{DATA}(l)$ and $CW_i(l)$ may alternatively be expressed as $t_{DATA}(j)$ and $CW_i(j)$, where j denotes the sending interface at node u). By defining $s_k(l)$ as the probability of delivering a frame through link l after k attempts, Draves et al. [1] approximated the expected delay $t(l)$ as:

$$\begin{aligned} t(l) &= \sum_{k=1}^7 s_k(l)t_k(l) = \sum_{k=1}^7 s_k(l) \sum_{i=1}^k (t_{DATA}(l) + CW_i(l)) \\ &= ett(l) + ebt(l) \end{aligned} \quad (1)$$

$$\begin{aligned} ett(l) &= t_{DATA}(l) \sum_{k=1}^7 s_k(l)k \approx t_{DATA}(l) \sum_{k=1}^{\infty} s_k(l)k \\ &= t_{DATA}(l)etx(l) \end{aligned} \quad (2)$$

$$ebt(l) = \sum_{k=1}^7 s_k(l) \sum_{i=1}^k CW_i(l) \quad (3)$$

The term $ebt(l)$ stands for *expected backoff time*, and represents the expected time spent in backoff. Similarly, $ett(l)$

²Intraflow (interflow) interference refers to the interference from the contention among nodes routing packets of the same (different) flow (flows).

stands for *expected transmission time*, and accounts for the expected time for sending a frame without considering the backoff delay. In Eq. (2), $ett(l)$ is simplified by assuming that the frame can be retransmitted as many times as needed, which allows to express it in terms of the *expected transmission count* (etx). For a link l , $etx(l) = \sum_{k=1}^{\infty} s_k(l)k$, and accounts for the expected number of transmissions to successfully send a frame across the link l [12]. The ett metric was proposed by Draves et al. [1] to estimate the total delay of a link. However, ett does not account for the backoff delay, which may have a considerable impact. To clarify this fact, decompose $CW_i(l)$ into $CW_i^{frz}(l)$, the time the sending interface freezes its backoff counter because another interface is transmitting in the same channel, and $CW_i^{dec}(l)$, the time the sending interface senses the channel idle and decrements its counter [14]. The contribution of $CW_i^{dec}(l)$ to $ebt(l)$ was proven in [1] to be minimal, and may be neglected. On the other hand, $CW_i^{frz}(l)$ is the term that implicitly accounts for the time other links on the same channel as l access the channel, and could be significant in medium or highly loaded scenarios. For $l = (u, v, c)$, define $\Gamma(l)$ to be the set of active radios in the interference range of u , v or both. Active radios are those interfaces contending for the use of the channel. Some authors simply classify a radio as active if the packets from that radio can be correctly received at the node of interest [20]. This approach, however, may erroneously classify a radio as active if that radio has currently no data to send and is not contending for the channel. To avoid this, we classify a radio as active if it has been sending data at a rate greater than a threshold δ^3 . Assuming that i) $CW_i^{frz}(l)$ is independent of i and denoted as $CW^{frz}(l)$, ii) $CW_i^{dec} \approx 0$ [1], and $etx(l) \approx \sum_{k=1}^7 s_k(l)k$, Eq. (3) may be rewritten as:

$$\begin{aligned} ebt(l) &\approx \sum_{k=1}^7 s_k(l) \sum_{i=1}^k CW^{frz}(l) \approx CW^{frz}(l)etx(l) \\ &\approx etx(l) \sum_{j \in \Gamma(l)} t_{DATA}(j) \end{aligned} \quad (4)$$

In Eq. (4), $CW^{frz}(l)$ is approximated as $\sum_{j \in \Gamma(l)} t_{DATA}(j)$, which is based on the fact that the time link l freezes its backoff counter is equivalent to the time the interfering radios are accessing the channel (the DCF [14] requires a sender attempting to send a frame to freeze its backoff counter while other nodes access the channel). Now, substituting Eqs. (2) and (4) into Eq. (1):

$$t(l) = etx(l)(t_{DATA}(l) + \sum_{j \in \Gamma(l)} t_{DATA}(j)) \quad (5)$$

Fig. 1 shows, for different link loss rates (i.e., probability that a frame transmission through a link is not successful) and different number of active nodes, i) the expected delay $t(l)$ given by Eq. (5), ii) the actual measured delay according to our simulations, and iii) the $ett(l)$ proposed in [1] as a delay metric. Our simulations were performed with the CNET

³To classify as active only those radios currently consuming channel resource, we set δ to $0.25r$, where r is the nominal rate the radio is operating at. We found this value suitable for the simulated scenario. However, δ may depend on the traffic pattern and may need to be set dynamically.

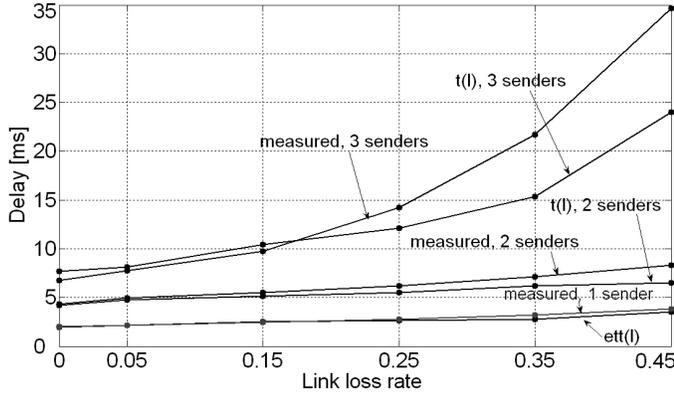


Fig. 1. Measured and estimated delays using ett and Eq. (5).

simulator [17] and involved simultaneous constant bit rate (CBR) data transfers of 200 packets per second, inducing interflow interference. The packet size was set to 1024 bytes. The simulations lasted 300 seconds, and 802.11a interfaces operating at 6 Mbps were used. The dataset of Fig. 1 represents the average delay of the received packets. As expected, $ett(l)$ matched very well the actual (measured) delay when there was only one sender, since $ebt(l) \approx 0$ and $t(l) \approx ett(l)$ (see Eq. (1)). However, as the number of senders is increased, the actual delay increases considerably, and the estimation given by Eq. (5) better matches the actual delay. The end-to-end delay of a path P can now be defined as:

$$D(P) = \sum_{l \in P} t(l) \quad (6)$$

C. Metric for path capacity

Our goal is to find channel-diverse paths, such that the intraflow interference is minimized and the path capacity maximized. Let $N(l)$ be the set of links interfering with link $l \in P$, including l . For a source node sending S -byte frames along P , we define the effective nominal rate of link l under intraflow interference as:

$$r'(l) = \frac{8S}{\sum_{l' \in N(l)} t_{DATA}(l')} = \frac{1}{\sum_{l' \in N(l)} 1/r(l')} \quad (7)$$

To illustrate the use of Eq. (7), consider Fig. 2, where the channel and the effective nominal rate r' in Mbps are shown over each link, and the nominal rate r is assumed to be 2 Mbps for all l . Hence, the first and third links in path $P1$, which happen to contend for channel c_1 , have an effective nominal rate of 1 Mbps. Similarly, $r'(l) = r(l)$ for all l in path $P2$, since there are no contending links in that path. Note that the computation of $r'(l)$ involves links in P only, and hence does not consider interflow; however, this was implicitly accounted for while computing the delay $t(l)$ of a link l , since Eq. (5)



Fig. 2. Impact of intraflow. The metric r' in Mbps and the channel are shown over each link. The nominal rate r is assumed to be 2 Mbps for all l .

considers the interfering set $\Gamma(l)$. Finally, the nominal capacity of a path P , considering intraflow interference, is defined as:

$$R(P) = \min_{l \in P} \{r'(l)\} \quad (8)$$

$R(P)$ may be viewed as the effective nominal rate of the bottleneck link in P and therefore should be maximized. By bottleneck link we refer to the link with minimum capacity considering r' as metric, which does take intraflow interference into account. This is a main difference to the approach where the bottleneck link with respect to the nominal capacity r only is considered (i.e., $R(P) = \min_{l \in P} \{r(l)\}$) and the contention among links in P is neglected (as in [13]).

IV. PROPOSED ALGORITHM

We start by showing how the algorithm operates on the topology of Fig. 3, where the effective nominal rate in Mbps is shown over each link, and the delay $t(l) = 2$ ms for all l . Suppose first we want to find the path from u to y with maximum capacity R and with delay exactly equal to 4 ms. Denote this path as $P_4^*(u, y)$. The way to get to y is through w or x ; so we need to compare these two paths:

$$R(P_4^*(u, y)) = \max \left\{ \begin{array}{l} \min\{R(P_{4-2}^*(u, w)), 6\}, \\ \min\{R(P_{4-2}^*(u, x)), 3\} \end{array} \right\}.$$

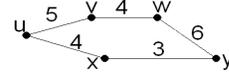


Fig. 3. A network with the effective nominal rate r' over each link.

The nominal capacity of both paths is found by evaluating Eq. (8). However, since there is no path $P_{4-2}^*(u, w)$ (i.e., path from u to w with a delay of 2 ms), then we only consider the path through x , which becomes the maximum capacity path. The path $P_{4-2}^*(u, x)$ can be similarly found. In general, finding the maximum capacity path whose end-to-end delay is less than a bound τ may be achieved by evaluating $P_d^*(u, y)$ for $d = 1, 2, \dots, \tau - 1$, and then choosing the one with maximum capacity. This idea is generalized as follows. Let $\hat{t}(l)$ be an scaled integer value representing the delay $t(l)$. The procedure to find $\hat{t}(l)$ is summarized at the end of this section. Let $P_d(u, v)$ be a path from u to v with delay d , and $P_d^*(u, v)$ the path with maximum capacity among all $P_d(u, v)$. Denote $P_d(u, v) \oplus l$ as the concatenation of $P_d(u, v)$ with link $l = (v, w, c)$ (i.e., $P_{d+\hat{t}(l)}(u, w) = P_d(u, v) \oplus l$), and $P_\tau^*(u, w)$ as the maximum capacity path from u to w with a delay bounded by τ . $P_\tau^*(u, w)$ is then determined by:

$$R(P_d^*(u, w)) = \max_{l=(v,w,c) \in E} \{R(P_{d-\hat{t}(l)}(u, v) \oplus l)\} \quad (9)$$

$$R(P_\tau^*(u, w)) = \max_{d < \tau} \{R(P_d^*(u, w))\} \quad (10)$$

Algorithm 1 iteratively solves Eq. (9) and then evaluates Eq. (10) to return a path or a set of paths (if multi-path is preferred), such that the end-to-end delay is bounded by τ . The running time of the algorithm is $O(\tau|V|^2)$. Note that the loop at line 13 is executed $O(1)$ times for a typical WMN⁴.

⁴Experimental results in [16] concluded that for optimal performance the number of neighbors for a given node should be less than 6.

Algorithm 1 MRA

```

1. INPUT:  $G(V, E)$ , source node  $v_s$ , destination node  $v_d$ , delay bound  $\tau$ .
2. OUTPUT: Set  $P_{v_d}^{v_s}$ 
3. /*Initialization */
4. for all pair  $u, w \in V$  and  $d \in \mathbb{Z}^+$ ,  $d < \tau$  do
5.    $P_d^*(u, w) = \text{NIL}$ 
6.   if  $\exists l = (u, w, c) \in E \mid \hat{t}(l) = d$  then
7.      $P_d^*(u, w) = l$  /*break ties by taking the  $l$  with highest  $r(l)$  */
8.   end if
9. end for
10. /*main loop*/
11. for  $d = 1$  to  $\tau - 1$  do
12.   for all pair  $u, w \in V$  do
13.     for all  $l = (v, w, c) \in E \mid d > \hat{t}(l)$  do
14.       if  $P_{d-\hat{t}(l)}^*(u, v) \neq \text{NIL}$  then
15.         if [ $P_d^*(u, w) = \text{NIL}$ ] or [ $R(P_d^*(u, w)) < R(P_{d-\hat{t}(l)}^*(u, v) \oplus l)$ ] then
16.            $P_d^*(u, w) = P_{d-\hat{t}(l)}^*(u, v) \oplus l$ 
17.         end if
18.       end if
19.     end for
20.   end for
21. end for
22. if multi-path option is desired then
23.    $P_{v_d}^{v_s} = \{ \text{Set of best link-disjoint paths in } \{P_d^*(v_s, v_d)\}, d = 1, 2, \dots, \tau - 1 \}$ 
24. else
25.    $P_{v_d}^{v_s} = \{ P_{\tau}^{**}(v_s, v_d) \}$  /* break ties by taking the path with lowest delay */
26. end if
27. return  $P_{v_d}^{v_s}$ 

```

The procedure for scaling and rounding the total delay $t(l)$ of a link l to $\hat{t}(l)$ is explained as follows. From Eq. (5), note that $t(l)$ depends on $etx(l)$, $t_{DATA}(l)$ and $t_{DATA}(j)$, $j \in \Gamma(l)$. We normalized the link delay to the minimum delay a link may experience when sending a data frame of size S_{ref} . For example, for 802.11b networks, the minimum delay is obtained when the following conditions hold: i) there is no interference ($\Gamma(l) = \emptyset$); ii) $r(l) = 11$ Mbps (maximum nominal rate); and iii) $etx(l) = 1$. Eq. (5) would then result in $t_{min} = S_{ref}8/(11 \text{ Mbps})$. The normalized value would finally be $\hat{t}(l) = \text{round}(t(l)/t_{min})$.

V. SIMULATION RESULTS

MRA was implemented as a distributed source-routed link-state protocol [1]⁵ in the CNET simulator [17], on top of the MAC-802.11 layer. For comparison purposes, we have also implemented the Dijkstra algorithm with delay (Eq. (5)), *wcett* [1], *hop count* and *rate cost* [13] metrics. We have chosen the CNET simulator due to its offered simplicity to extend the protocol stack and the wireless module as well as its multi-radio multi-channel support. In addition, its propagation model allows to employ not only the Friis free-space model but also other models defined by the user, which permits the use of more realistic scenarios (we have used the two-ray model⁶). Interference is modeled by the protocol model of interference [19]. The interfaces are assumed to be 802.11a omnidirectional radios operating at fixed nominal rates, which are randomly selected from the standard values⁷. Nodes are equipped with two or three radios tuned to fixed channels; we assume the

⁵In distributed source-routed link-state routing protocols, nodes propagate link information such as delay, capacity, and channel assignment information. Then, based on this topology information, the routing algorithm at each node finds paths for a given destination. For more details about the implementation and overhead of this kind of protocol, please refer to [1].

⁶For the two-ray model and a transmission power of 30 mW used in the simulations, the transmission and interference ranges are 133 and 373 meters.

⁷The standard does not specify how nominal rates should be selected. We have designed MRA to be independent of the selection mechanism.

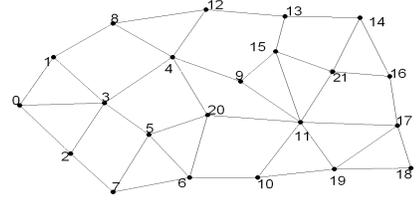


Fig. 4. Network topology used for simulation purposes. A link between two nodes implies they have at least one interface tuned to the same channel, and they are within the transmission range of each other.

existence of five channels (c_1 to c_5). The application layer generated CBR flows of 1024-byte packets, and each simulation result was averaged over 10 runs. The evaluation metrics are the *end-to-end delay* and the *packet delivery ratio* (PDR). PDR refers to the ratio of the number of packets received by the destination node to the number of packets generated by the source node, and it reflects the capacity of a path used to route packets⁸. MRA was tested under two scenarios⁹, performed on the network topology shown in Fig. 4. The first scenario tested the algorithm's performance under multiple flows and tightly restricted the end-to-end delay to predefined values. MRA was then compared with shortest delay path algorithm (SP-D), which minimizes Eq. (6). The second scenario focused on the algorithm's performance under high traffic loads and relaxed the end-to-end delay to more loosely values. MRA was then compared with approaches that do not guarantee a delay bound but attempt to optimize throughput.

A. Scenario I

We simulated 6 concurrent flows with the following source-destination pairs: 8-11, 13-5, 9-10, 6-1, 4-0, 11-16. Each source generated 100 packets per second. Flow 1 started at $t = 1$ min. The inter-arrival time between flows was set to 1 min; the simulation ended at $t = 7$ min, and τ was set to 5 ms. Figs. 5(a) and 5(b) show the per-flow results in terms of PDR and end-to-end delay. MRA delivered almost 100 % of the packets of flows 1, 4, 5 and 6; and about 80 % and 93 % of the packets of flows 2 and 3. On the other hand, the PDR for SP-D is below 0.8 for flows 1, 2, 3 and 6. The delay of MRA, though mostly higher than the end-to-end delay of SP-D, was kept below τ , except for some few packets of flows 1 and 2. The main reason of the higher capacity of paths found by MRA is the diversity of channels used along those paths.

B. Scenario II

We again simulated the same 6 flows as in scenario I, but in saturated case, where source nodes always have packets to send. Although MRA has been designed to find delay-bounded paths, the idea of this test is to relax the delay bound and compare MRA with SP-wcett [1], SP-cost [13], and SP-hop. SP-wcett minimizes a weighted sum of two terms quantifying delay and intraflow. In SP-cost, a link cost is the inverse of its nominal rate; then, the routing algorithm minimizes the

⁸Given that the PDR is proportional to the per-destination throughput (bps at the receiver), a throughput analysis can be also inferred from the PDR.

⁹For further scenarios, please refer to [10].

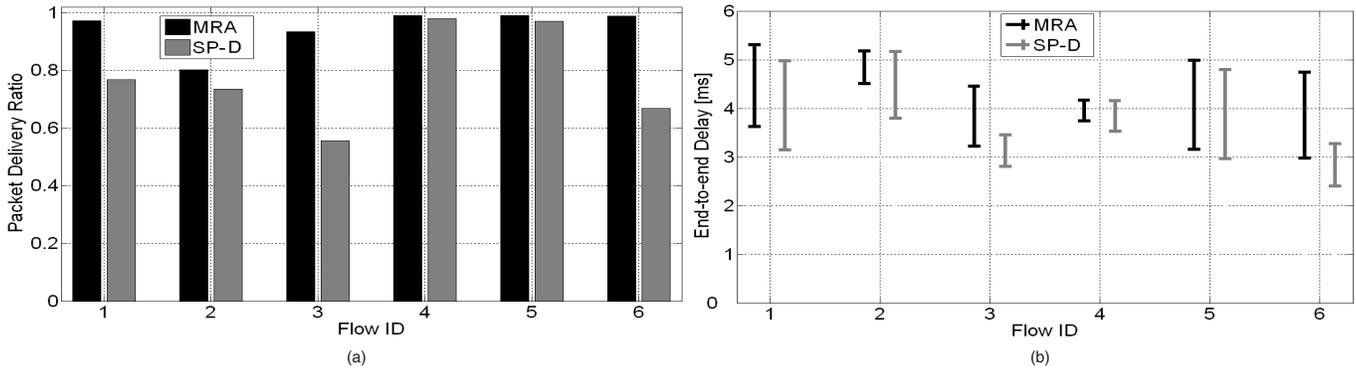


Fig. 5. Packet delivery ratio and end-to-end delay for scenario I.

maximum among the cost of the links along the path. SP-hop minimizes the number of hops to the destination. Since neither of these approaches can find delay-bounded paths, we only compared MRA with them in terms of PDR. We relaxed the value of τ to 3 times the delay of the path with shortest delay (i.e., firstly, the delay of the shortest path is found. Then, τ is defined as 3 times that value). Table I shows the ratio of the PDR values normalized to the PDR obtained by MRA. While SP-wcett delivered about 93% of the packets delivered by MRA, SP-cost and SP-hop delivered 86% and 81%. One reason of the better performance of MRA is that it considers the interference from interfering radios while computing link delays (Eq. (5)), which enforces the minimization of interflow interference. In addition, the load-balancing scheme contributes to the performance of MRA. We also noted the good performance of SP-wcett; the difference between MRA and SP-wcett relative to SP-cost and SP-hop is that the former approaches consider intraflow interference. SP-hop, as expected, had the lowest PDR, because it takes neither interference nor nominal rates into account.

Table I. Normalized PDR - Scenario II.

SP-wcett [1]	SP-cost [13]	SP-hop
0.9345	0.8671	0.8095

VI. CONCLUSION

In this paper, we have presented a routing algorithm for WMNs to find paths with high capacity while bounding the end-to-end delay to a desired value. Through the dynamic programming model presented in the paper, both the path capacity and the end-to-end delay metrics can be simultaneously considered, providing an alternative approach to the traditional weighted sum technique. Additionally, our proposed algorithm provides the load-balancing through link-disjoint paths option.

To measure link delays, we have extended the procedure proposed in [1] to consider the backoff delay when multiple nodes contend for the channel. We have also defined a metric for path capacity that takes intraflow interference and nominal rate of links into account. Simulation results show that MRA can find delay-bounded paths with higher capacity than other approaches, while splitting the traffic through multiple paths.

As a future work, we plan to extensively simulate MRA under different network topologies and to capture the impact

that parameters such as data rate and transmission power have over the algorithm. Incorporating scheduling is an interesting research direction which we plan to pursue. For link scheduling, a conflict-free MAC protocol to jointly operate with MRA may be needed to improve the coordination among nodes, which is a fundamental cause of unfairness of the DCF [21].

REFERENCES

- [1] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, MobiCom 2004, Philadelphia, PA - USA.
- [2] M. Kodialam, T. Nandagopal, Characterizing the capacity region in multi-radio multi-channel wireless mesh networks, MobiCom 2005, Germany.
- [3] Y. Yang, J. Wang, R. Kravets, Designing routing metrics for mesh networks, WiMesh 2005, Santa Clara, CA - USA.
- [4] W. Zhou, D. Zhang, D. Qiao, Comparative study of routing metrics for multi-radio multi-channel wireless networks, WCNC 2006, Las Vegas.
- [5] L. Ma, M. Denko, A routing metric for load-balancing in wireless mesh networks, AINA 2007, Ontario - Canada.
- [6] K. Deb, Multi-Objective optimization using evolutionary algorithms, Wiley, July 2001, pp. 50-52.
- [7] S. Dasgupta, C. Papadimitriou, U. Vazirani, Algorithms, Mc Graw Hill, 2008, p. 156.
- [8] FCC Report and Order: FCC-03-324, February 2004.
- [9] G. Marfia, G. Pau, E. De Sena, E. Giordano, M. Gerla, Evaluating vehicle network strategies for downtown portland: opportunistic infrastructure and the importance of realistic mobility models, MobiSys 2007, PR.
- [10] J. Crichigno, J. Khoury, M. Y. Wu, W. Shu, A dynamic programming approach for routing in wireless mesh networks, Technical Report ECE-TR-08-009, Aug. 2008. <http://www.ece.unm.edu/~jkhoury/meshTR.pdf>
- [11] J. So, N. Vaidya, Load-balancing routing in multichannel hybrid wireless networks with single network interface, IEEE Transactions on Vehicular Technology, Vol. 55, Issue 3, 2006, pp. 806 - 812.
- [12] D. Couto, D. Aguayo, J. Bicket, R. Morris, High-throughput path metric for multi-hop wireless routing, MobiCom 2003, San Diego, CA - USA.
- [13] L. Iannone, K. Kabassanow, S. Fdida, Evaluation of cross-layer rate-aware routing in a wireless mesh network test bed, Eurasip journal on wireless communications and networking, vol 2007, Article Id 86510.
- [14] IEEE Std. 802.11, Wireless lan medium access control (MAC) and physical layer (PHY) specifications, 1999.
- [15] K. Xu, M. Gerla, S. Bae, Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks, Ad Hoc Network, 1(1), July 2003.
- [16] E. Belding, P. Melliar-Smith, L. Moser, An analysis of the optimum node density for ad hoc mobile networks, ICC 2001, Helsinki, Finland.
- [17] <http://www.csse.uwa.edu.au/cnet3/>
- [18] Z. Wu, S. Ganu, I. Seskar, D. Raychaudhuri, Experimental investigation of PHY layer rate control and frequency selection in 802.11-based ad-hoc networks, SIGCOMM 2005, Philadelphia, PA - USA.
- [19] P. Gupta, P. Kumar, The capacity of wireless networks, IEEE Transactions on Information Theory, Vol. 46, No. 2, pp. 388-404, 2000.
- [20] K. Ramachandran, E. Belding, K. Almeroth, M. Buddhikot, Interference-aware channel assignment in multi-radio wireless mesh networks, Infocom 2006, Barcelona - Spain.
- [21] M. Garetto, T. Salonidis, E. Knightly, Modeling per-flow throughput and capturing starvation in csma multi-hop wireless networks, Infocom 2006, Barcelona - Spain.