Workload-Aware Opportunistic Routing in Multi-Channel, Multi-Radio Wireless Mesh Networks

Fan Wu
Department of Computer Science and Engineering
Shanghai Key Laboratory of Scalable Computing and Systems
Shanghai Jiao Tong University
fwu@cs.sjtu.edu.cn

Nitin Vaidya
ECE & Coordinated Science Lab
University of Illinois at Urbana-Champaign
nhv@illinois.edu

Abstract—**Opportunistic routing emerged as a novel technique to cope with the problem of highly unpredictable and lossy wireless channels in urban wireless mesh networks. However, existing opportunistic routing protocols only consider single-radio wireless nodes, and assume that all the nodes work on the same channel, without exploiting possible concurrent transmissions by multi-radio nodes over orthogonal channels provided by IEEE 802.11 protocols. Examples show that simply integrating existing channel assignment schemes and the opportunistic routing technique may not achieve satisfactory system performance. In this paper, we present WACA, which is a Workload-Aware Channel Assignment algorithm for opportunistic routing in multi-channel, multi-radio wireless mesh networks. Evaluation results show that WACA always achieves highest average throughput among the evaluated algorithms, and its median throughput is at least 16.1% higher than the compared ones.**

I. INTRODUCTION

Wireless mesh networks provide an alternative way to deploy broadband network infrastructures to local communities at low cost [1]–[4]. However, the deployment of wireless mesh networks has a major challenge, which is throughput scalability. Due to the highly unpredictable and lossy wireless channels, the throughput achieved by traditional deterministic routing protocols in wireless mesh networks can be quite poor. This problem is particularly serious in urban areas, where exist many sources of interference from various wireless applications [2], [5].

To cope with the highly unpredictable and lossy wireless channels, opportunistic routing emerged as a novel technique to allow any node that overhears the packet to participate in packet forwarding, which is different from the traditional deterministic routing techniques. In an early work, Biswas and Morris [6] introduced the ExOR opportunistic routing protocol and showed that it can achieve superior end-to-end throughput than the traditional deterministic forwarding. Recently, Chachulski et al. [7] proposed the MORE opportunistic routing protocol to address issues in ExOR and achieve even higher throughput in wireless mesh networks.

However, existing opportunistic routing protocols only consider single-radio wireless nodes, and assume that all the nodes work on the same channel, without exploiting possible concurrent transmissions by multi-radio nodes over orthogonal channels provided by IEEE 802.11 protocols (3 orthogonal channels in 802.11b/g and 12 in 802.11a). Although a considerable amount of work has been done on multi-channel, multi-radio assignment in wireless mesh networks (*e.g.*, [8]–[13]), simply integrating the existing channel assignment schemes and the opportunistic routing technique may not produce a satisfactory result. Our previous work [14] demonstrates that a carefully designed channel assignment may achieve much higher throughput than traditional channel assignment, when opportunistic routing technique is provided. Therefore, it is highly needed to design new channel assignment algorithms for opportunistic routing.

However, designing a good channel assignment algorithm for opportunistic routing is not a trivial task. One of the major challenges, which is not limited to channel assignment problem for opportunistic routing but applies to channel assignment problem in general, is the computation complexity. It is shown that the problem of finding the optimal channel assignment is NP-complete [15], [16]. Another major challenge is the tradeoff between opportunistic throughput gain and multi-channel throughput gain. On the one hand, the opportunistic routing improves throughput by letting all downstream nodes stay on the same channel as the sender, which maximizes the probability of a packet being received by at least one of the downstream nodes. The forwarders again compete for the same channel with their upstream nodes to forward the overheard packets. Unfortunately, contentions from multiple nodes may significantly decrease the goodput of a channel [17]. On the other hand, the multi-channel routing boosts throughput by distributing nodes/radios onto different channels, such that simultaneous transmissions are enabled between interfering nodes and the average level of contention is reduced. But it also decreases the opportunity of a packet being heard by downstream nodes. Therefore, finding a good tradeoff between these two "conflicting" techniques is essential for designing a channel assignment algorithm for opportunistic routing.

In this paper, we present WACA, which is a Workload-Aware Channel Assignment algorithm for opportunistic routing in multi-channel, multi-radio wireless mesh networks. Intuitively, the algorithm identifies the nodes with high workloads in a flow as bottlenecks, and tries to assign channels to

these nodes with high priority. The major contributions of this paper are as follows:

- We present a simpe extension for the opportunistic routing protocol MORE to work in multi-channel, multi-radio wireless mesh networks, namely EMORE.
- We propose a novel workload-aware channel assignment algorithm (WACA), which computes both a channel assignment and a routing strategy for running EMORE.
- We extensively evaluate WACA's performance, and compare it with existing channel assignment algorithms. Numerical results show that WACA significantly improves the throughput.

The rest of this paper is organized as follows. In Section II, we present our network model, formulate the problem, and briefly review the opportunistic routing protocol MORE. In Section III, we describe a extended version of MORE for multi-channel, multi-radio wireless mesh networks. In Section IV, we present our workload-aware channel assignment algorithm. In Section V, we report the evaluation results. In Section VI, we discuss a related issue. In Section VII, we briefly review the related works. Finally, we conclude the paper and point out potential future works in Section VIII.

II. TECHNICAL PRELIMINARIES

In this section, we present our network model and assumptions, and formulate the channel assignment problem for wireless mesh networks. We then briefly review an efficient opportunistic routing protocol — MORE.

A. Network Model and Assumptions

We consider a wireless mesh network with a set N of stationary wireless nodes (routers), where each node is equipped with r radio interfaces. Let K denote the set of orthogonal (non-interfering) and homogenous channels. For simplicity, we assume that all the nodes use the same transmission rate over their radios, and we normalize the transmission rate as a unit constant.

We assume that there is no power control scheme. Let $\epsilon_{i,j}$ be the link loss probability from node i to node j on any channel; that is, if a packet is transmitted from node i to node j on a common channel shared by them, then with probability $\epsilon_{i,j}$ the packet cannot be decoded. It is also interesting to study the case, in which the link loss probabilities are different on available channels. However, we focus on homogenous channels in this paper, and left the problem of dynamic and diverse channels in our future work.

For simplicity, we do not consider the throughput loss caused by nodes' contention for communication medium. We also do not consider the hidden terminal problem, which has not been fully resolved in exist opportunistic routing protocol (e.g., [6], [7], [18]–[23]).

B. Problem Formulation

Given a static wireless mesh network of router nodes with multiple radio interfaces, we wish to assign one or multiple channels to each node, such that the number of different channels assigned to a node is not more than the number of radios on the node. The objective of the channel assignment problem for wireless mesh network is to maximize the throughput between a source node and a destination node.

Formally, the problem of static channel assignment for a multi-radio wireless mesh network over a set of N nodes, given a set of K channels, is to compute a function $f : N \rightarrow \mathcal{P}(K)$, to maximize the throughput between a given source-destination pair (src, dst).

The above problem formulation does not specify the routing protocol. In this paper, we assume opportunistic routing protocol (e.g., MORE) is used.

C. Opportunistic Routing Protocol — MORE

Opportunistic routing is an emerging technique to achieve high throughput despite lossy wireless links. Instead of deterministically choosing the next hop before transmitting a packet, opportunistic routing allows multiple nodes that overhear the packet to participate in forwarding. MORE is a representative and efficient opportunistic routing protocol. Our study in this work is based on MORE.

MORE is designed for a single-radio, single-channel setting. In Section III, we extend it to a multi-radio, multi-channel setting.

Let the "distance" from a node i to the destination be the expected number of transmissions to deliver a packet from node i to the destination, i.e., ETX [24] and EOTX [25]. For any two nodes, i and j, let $i < j$ denote that node i is closer to the destination than node j.

Source Node: The source node of a session divides its traffic into a number of batches, where each batch consists of B packets. When the 802.11 MAC is ready to send, the source node generates a random linear combination of the packets in the current batch and broadcasts the coded packet. Each coded packet has a packet header containing sufficient information for routing. It stops the transmission of a batch after acknowledged by the destination, and proceeds to the next one.

Intermediate Node: When an intermediate node hears a packet from an upstream node, the contents of this packet (including the header) decide whether this intermediate node is triggered to transmit a packet.

Especially, in MORE, each of the intermediate nodes keeps a credit counter. When an intermediate node i receives a packet from an upstream node, it increments the credit counter by $credit_i$, which is the number of transmissions that a node should make for every packet it receives from a node farther from the destination in the EOTX metric:

$$credit_i = \frac{z_i}{\sum_{j > i} z_j (1 - \epsilon_{j,i})}, \qquad (1)$$

where z_i is the expected number of transmissions that node i should make for delivering one packet from source to destination. If the credit counter is positive, the node creates a coded packet (which is a random linear combination of the innovative coded packets heard from the same batch), broadcasts it, and then decrements the credit counter.

Destination Node: The destination uses the contents of its received packets to decide whether it has sufficient information for decoding. If so, it decodes the packets in this batch and

sends an acknowledgment using a traditional best path routing protocol.

III. EXTENDING MORE TO MULTI-CHANNEL, MULTI-RADIO SETTING

As we have mentioned, MORE was originally designed for a single-radio, single-channel setting. In this section, we describe a simple extension for MORE (EMORE) to work in a multi-radio multi-channel setting.

A. EMORE

We allow nodes that are equipped with multiple radio interfaces to work on multiple channels simultaneously. We assume that there is no throughput gain when a node has more than one radio on the same channel. Therefore, we require that every node should tune at most one radio on a channel. Now, let's assume that there is a channel assignment. (We will present an algorithm to compute such a channel assignment in Section IV.) It is possible that the number of channels assigned to a node in a channel assignment be less than the number of radios on that node, in which case the node can use redundant radios to serve other flows in the network. However, we focus on the throughput of a single flow in this work. The problem of maximizing the total throughput of multiple flows will be considered in our future work.

Every node can use its assigned channels for packet transmission and reception. Let $X_i^k \in \{0, 1\}$ denote whether a radio of node $i \in N$ is assigned to channel $k \in K$[1].

$$X_i^k = \begin{cases} 1 & \text{if a radio of node } i \text{ is assigned to channel } k, \\ 0 & \text{otherwise.} \end{cases}$$

Given $\mathcal{X} = \{X_i^k | i \in N, k \in K\}$, the expected number of packets D_i that node i need to forward for delivering one packet from source to destination is:

$$D_i = \sum_{k \in K} D_i^k, \tag{2}$$

where D_i^k is the expected number of packet received by node i but not received by any of its downstream nodes on channel k from upstream nodes:

$$D_i^k = X_i^k \sum_{j > i} \left(Z_j^k (1 - \epsilon_{j,i}^k) \prod_{h < i} \left(1 - X_h^k (1 - \epsilon_{j,h}^k) \right) \right). \tag{3}$$

Here, Z_j^k is the expected number of transmissions that node j makes on channel k, for delivering one packet from source to destination. We call D_i^k and Z_i^k node i's duty and workload on channel k. Let $\mathcal{D} = \{D_i^k | i \in N, k \in K\}$, and $\mathcal{Z} = \{Z_i^k | i \in N, k \in K\}$.

Since a node may be assigned multiple channels for transmitting packets, we split the node's total duty to different channels, such that

$$D_i = \sum_{k \in K} X_i^k L_i^k, \tag{4}$$

where L_i^k is node i's workload on channel k. L_i^k is a valid workload only when node i has allocated a radio on channel k, i.e., $X_i^k = 1$.

[1]We do not distinguish which radio is assigned to channel k.

Thus, the workload of node i on channel k is:

$$Z_i^k = \frac{X_i^k L_i^k}{1 - \prod_{j < i} \left(1 - X_j^k (1 - \epsilon_{i,j}) \right)}. \tag{5}$$

Instead of keeping a single credit counter, we maintain $|K|$ credit counters for each of the nodes. Each of the credit counters corresponds to a distinguished channel. Let $credit_i^k$ be the number of transmissions on channel k that an intermediate node i should make for every packet it receives from an upstream node:

$$credit_i^k = \frac{Z_i^k}{\sum_{k \in K} \left(X_i^k \sum_{j > i} \left(Z_j^k (1 - \epsilon_{j,i}) \right) \right)}. \tag{6}$$

If the credit counter on channel k becomes positive, the node creates a coded packet, broadcasts it on channel k, and then decrements the credit counter.

B. Throughput Estimation

In this section, we propose a simple way to estimate the throughput of EMORE. The estimation only serves as a tool to help us to compute the channel assignment in Section IV.

We assume that node i's expected transmission rate on channel k is proportional to Z_i^k among the nodes who are sharing the communication media with it. To avoid collision, when a node transmits a packet, the other nodes, which may interfere with the reception of the packet, should keep silent. Therefore, we define the set of conflicting nodes of node i as $\mathcal{F}_i$:

$$\mathcal{F}_i = \{j | \epsilon_{i,j} < 1 \vee \epsilon_{j,i} < 1 \vee (\exists k \in N, \epsilon_{i,k} < 1 \wedge \epsilon_{j,k} < 1)\}. \tag{7}$$

Noting that some of the channels may not be saturated during the transmission, we introduce a variable, λ_i^k, to indicate the effective usage ratio of channel k by node i's conflicting set. Let $\Lambda = \{\lambda_i^k | i \in N, k \in K\}$.

Then, the normalized total effective transmission rate of node i on all channels, denoted by T_i, is:

$$T_i = \sum_{k \in K} T_i^k, \tag{8}$$

where T_i^k is the normalized effective transmission rate of node i on channel k:

$$T_i^k = \lambda_i^k \cdot \frac{X_i^k Z_i^k}{\sum_{j \in \mathcal{F}_i} X_j^k Z_j^k}. \tag{9}$$

Finally, the end-to-end throughput can be calculated as follows:

$$Throughput = \sum_{k \in K} \left(X_{dst}^k \sum_{i \in N} T_i^k (1 - \epsilon_{i,dst}) \right). \tag{10}$$

We note that the calculation of end-to-end throughput is based on the assumption that the hidden terminal problem is fully resolved by some MAC layer coordination scheme. We also note that the bandwidth overhead for delivering the acknowledgements is very small compared with the data transmitted. Therefore, we ignore this overhead when calculating

the throughput. However, the calculations shown above can serve as a tool to help us to derive a good channel assignment.

In the above equations, there are three set of variables need to be computed: the channel assignment $\mathcal{X}$, the set of workloads $\mathcal{Z}$, and the set of effective channel usage ratio Λ. The other variables can be derived from them. In section IV, we will present algorithms to compute them.

IV. WORKLOAD-AWARE CHANNEL ASSIGNMENT ALGORITHM

In this section, we present our workload-aware channel assignment algorithm (WACA) for opportunistic routing. Our algorithm is composed of three major modules:

1) Workload-aware channel assignment: Compute a channel assignment $\mathcal{X}$ based on nodes' workloads.
2) Workload distribution: Given a channel assignment $\mathcal{X}$, compute a workload distribution $\mathcal{Z}$ for the nodes on the assigned channels.
3) Throughput computation: Given a channel assignment $\mathcal{X}$ and nodes' workloads $\mathcal{Z}$ over the channels K, compute the maximal throughput and the set of effective channel usage ratios Λ.

In subsequent sections, we will describe each module in detail. For ease of explanation, we begin with the throughput computation module.

A. Throughput Computation

This module computes the optimal effective channel usage ratios Λ based on the channel assignment $\mathcal{X}$ and nodes' workload distribution $\mathcal{Z}$, such that the throughput is maximized.

We formulate this problem as a linear program. The objective is to maximize the throughput, which is also the reception rate at the destination:

$$Maximize \qquad \sum_{k \in K} \left(X_{dst}^k \sum_{i \in N} T_i^k (1 - \epsilon_{i,dst}) \right)$$

Subject to:

$$\sum_{k \in K} \left(X_i^k \sum_{j > i} \left(T_j^k (1 - \epsilon_{j,i}^k) \prod_{h < i} \left(1 - X_h^k (1 - \epsilon_{j,h}^k) \right) \right) \right)$$
$$= \sum_{k \in K} \left(T_i^k \left(1 - \prod_{h < i} \left(1 - X_h^k (1 - \epsilon_{i,h}^k) \right) \right) \right),$$
$$\forall i \in N - \{src, dst\} \qquad (11)$$

$$0 \le \lambda_i^k \le 1, \qquad \forall i \in N, \forall k \in K \qquad (12)$$

Here constraint (11) indicates flow conservation — the amount of effective incoming flow is equal to that of effective outgoing flow for every node except the source and the destination. Constraint (12) indicates that the effective channel usage ratios should be in the range of $[0,1]$. For simplicity, we do not re-list constraints (9) here, which ensures the proportional relation between the normalized effective transmission rate of a node and its workload on each channel. The module of throughput computation requires $O(|K|^{3.5}|N|^{3.5})$ operations, when interior point methods [26] are used.

B. Workload Distribution

The workload distribution module computes a workload distribution $\mathcal{Z}$ for the nodes on the assigned channels, given a channel assignment $\mathcal{X}$.

We observe that maximal throughput is usually achieved when the nodes' workloads are evenly distributed over the used channels.

We model the problem of computing the workload distribution as a convex nonlinear program, which can be solved numerically very efficiently. The program tries to evenly distribute workloads onto the channels. We define the workload on a channel $k \in K$ be the sum of the nodes' workloads on this channel:

$$W_k = \sum_{i \in N} X_i^k Z_i^k. \qquad (13)$$

Therefore the objective is to minimize the standard deviation of channels' workloads:

$$Minimize \qquad \sqrt{\frac{1}{|K|} \sum_{k \in K} (W_k - \overline{W})^2}$$

where $\overline{W}_i$ is the mean of channel workloads

$$\overline{W}_i = \frac{\sum_{k \in K} W_k}{|K|}.$$

Subject to constraint (2), (3), (4), (5), and:

$$D_{src} = 1 \qquad (14)$$

$$Z_i^k \ge 0, \qquad \forall i \in N, \forall k \in K \qquad (15)$$

Here constraint (14) states that the workload of the source node is 1. Constraint (15) ensures that every node has non-negative workload on each of the channels. The workload distribution module also requires $O(|K|^{3.5}|N|^{3.5})$ operations, when interior point methods are used.

C. Workload-Aware Channel Assignment

The most important component is the workload-aware channel assignment module, which interacts with workload distribution module and throughput computation module, and greedily assigns channels to the nodes. Noting that a node with higher workload is more likely to be a bottleneck, we propose an algorithm, which tries to assign more channels to higher workload nodes.

Because the channel assignment depends on nodes' workloads (generally nodes with higher workloads should be assigned with more channels), and nodes' workload depends on the channel assignment, there is a circular dependency between channel assignment and nodes' workloads. To break this circularity, we start by assigning every node a default channel, and iteratively improve the throughput by greedily assigning channels based on nodes' per assigned channel workloads and then revoking under-utilized channels. Finally, the computed channel allocation together with the workload distribution will be delivered to each forwarding node through a common control channel.

Algorithm 1 shows the pseudo-code of our workload-aware channel assignment algorithm. In lines 1-3, the algorithm initializes the channel assignment matrix $\mathcal{X}$ by assigning

Algorithm 1 Workload-Aware Channel Assignment Algorithm

Input: A set of nodes N, a set of equipped radios R, a set of channels K, and a set of link loss probabilities.

Output: A channel assignment $\mathcal{X}$.

1: **for all** $i \in N$ **do**
2:    $X_i^1 \leftarrow 1$.
3: **end for**
4: $\mathcal{Y} \leftarrow \mathcal{X}$.
5: $(\mathcal{Z}, \mathcal{D}) \leftarrow ComputeWorkloads(\mathcal{X})$.
6: $throughput \leftarrow ComputeThroughput(\mathcal{X}, \mathcal{Z})$.
7: $increment \leftarrow throughput$.
8: **while** $increment > 0$ **do**
9:    **while** $\exists i \in N, r_i > \sum_{k \in K} Y_i^k \wedge r_i < |K|$ **do**
10:      $i \leftarrow \underset{i \in N \wedge r_i > \sum_{k \in K} Y_i^k}{argmax}(Z_i / \sum_{k \in K} Y_i^k)$.
11:      **if** $i \neq dst$ **then**
12:        $k \leftarrow \underset{k \in K \wedge Y_i^k = 0}{argmin}\left(\sum_{j \in N} Z_j^k\right)$.
13:        $Y_i^k \leftarrow 1$.
14:        **for all** $j \in N, j < i \wedge \epsilon_{ji} < 1 \wedge r_j > \sum_{k \in K} Y_j^k$ **do**
15:           $Y_j^k \leftarrow 1$.
16:        **end for**
17:      **else**
18:        $k \leftarrow \underset{k \in K \wedge Y_i^k = 0}{argmax}\left(\sum_{j > i} Y_j^k (1 - \epsilon_{j,i}) D_j\right)$.
19:        $Y_i^k \leftarrow 1$.
20:      **end if**
21:      $(\mathcal{Z}, \mathcal{D}) \leftarrow ComputeWorkloads(\mathcal{Y})$.
22:    **end while**
23:    $throughput' \leftarrow ComputeThroughput(\mathcal{Y}, \mathcal{Z})$.
24:    $increment \leftarrow throughput' - throughput$.
25:    **if** $increment > 0$ **then**
26:      $\mathcal{X} \leftarrow \mathcal{Y}$.
27:    **end if**
28:    **for all** $i \in N$ **do**
29:      **for all** $k \in K$ **do**
30:        **if** $D_i^k < \alpha$ and $Z_i^k < \beta$ **then**
31:           $Y_i^k \leftarrow 0$.
32:        **end if**
33:      **end for**
34:    **end for**
35: **end while**
36: **return** $\mathcal{X}$.

channel 1 as the default channel to each of the nodes. In line 4, the algorithm copies $\mathcal{X}$ to $\mathcal{Y}$, which is a tentative channel assignment used in the iterations later. Then, in lines 5-6, the algorithm calls the workload distribution module (presented in Section IV-B) and the throughput computation module (presented in Section IV-A), to calculate the nodes' workloads and the estimated throughput on the default channel assignment, respectively. When calling the workload distribution module, we also ask the module to return the set of nodes' duties $\mathcal{D}$, which will assist us to prune the channel assignment later. The algorithm also initializes the variable $increment$ by the initial throughput.

Next, Algorithm 1 iteratively updates the channel assign-

ment $\mathcal{X}$ to $\mathcal{Y}$, until $\mathcal{Y}$ fails to achieve a higher throughput. In particular, each iteration is composed of two major procedures:

- Greedy channel assignment (lines 9-27): In the greedy channel assignment procedure, the algorithm iteratively and greedily assigns channel to the nodes. The operations in this procedure are on the tentative channel assignment $\mathcal{Y}$. In each iteration, the algorithm checks whether there is any node with free radio. If yes, it finds the node i with the heaviest per assigned channel workload among the nodes with free radio (line 10)[2]. If the node i is not the destination, the algorithm assigns the channel k, with the lightest channel workload[3], to the node i and its one-hop downstream nodes (lines 12-16). If the node i is the destination, it means that all the nodes except the destination have been assigned channel, because the destination has the smallest workload and EOTX. The algorithm assigns to node i the channel that has the largest weighted duty, where the weights are virtual links' packet reception probabilities $(1 - \epsilon_{j,i})$ (lines 18-19). This assignment may potentially increase the throughput at the last hop. At the end of each iteration, workload distribution module is called to recalculate nodes' workloads. Finally, when all the nodes' radios are used in the tentative channel assignment $\mathcal{Y}$, the algorithm stops the iteration for greedy channel assignment.

  The algorithm now computes the throughput achieved by the tentative channel assignment $\mathcal{Y}$ (line 23). If a higher throughput is reached, it updates the channel assignment $\mathcal{X}$ to $\mathcal{Y}$ (lines 24-27).

- Channel assignment pruning (lines 28-34): In the channel assignment pruning procedure, the algorithm removes the channel assignment item, on which both the duty and the workload are less than their thresholds (α for duty and β for workload). Intuitively, if a node's radio does not contribute (receive or send) much on current channel, it should be tuned to other channels that are beneficiary to the flow.

Finally, the algorithm return a channel assignment $\mathcal{X}$.

Complexity: Both the workload distribution and throughput computation modules require $O(|K|^{3.5}|N|^{3.5})$ operations, when interior point methods [26] are used. In Algorithm 1, the outer loop is executed at most $r|N|$ times, and for each iteration, the inner loop is also executed at most $r|N|$ times. Each iteration in the inner loop requires $O(|K|^{3.5}|N|^{3.5})$ operations. Consequently, the Algorithm 1 requires $O(r^2|K|^{3.5}|N|^{5.5})$ operations.

V. EVALUATION

We evaluate WACA using randomly generated wireless networks, and compare its performance with existing channel assignment algorithms.

A. Methodology

We compare the throughput of WACA with the following 5 schemes:

---

[2]If there is a tie, the node with larger EOTX is selected.
[3]If there is a tie, the channel with the least number of radios is selected.

- TABU [13]: This is a tabu-based centralized channel assignment algorithm. When OR is not specified, shortest path routing protocol is used to find the route from source to destination.
- TABU+OR: TABU channel assignment with the multi-channel opportunistic routing protocol EMORE explained in Section III. In this case, the module of workload distribution and throughput computation are also applied.
- RAND: Random channel assignment with shortest path routing.
- RAND+OR: Random channel assignment with EMORE.
- UNIFORM+OR: Uniformly allocating the same set of channels to the nodes, and EMORE is used for routing. The performance of this case is identical to that of simply applying MORE to multiple channels simultaneously. Since it is already shown that MORE achieve much higher throughput than shortest path routing in the literature, we only consider uniform channel allocation with EMORE.

We perform two set of evaluations. In the first set of evaluations, we randomly distribute 25 wireless nodes in a terrain area of 1000 meters × 1000 meters; while in the second set of evaluations, we fix the terrain area at 750 meters × 750 meters, and randomly distribute 4, 9, 16, and 25 into it. In each run, we examine WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR sequentially between the same source-destination pair. The source is always backlogged. We list the parameters used to obtain numerical results in Table I. We note that similar results can be gotten when higher bit rates are used. The linear and nonlinear programs are solved by LINDO API [27].

TABLE I
PARAMETERS USED TO OBTAIN NUMERICAL RESULTS

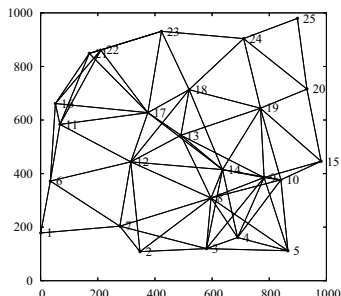|	
Antenna Height	1 m
TX Power	15 dBm
Noise Figure	10
Pathloss Model	TWO-RAY model
Packet Length	1500 bytes
Channel Bit Rate	1 Mbps
α	0.05
β	0.05

B. End-to-End Throughput

Fig. 1.   Topology of the Random Generated Network.

Our first set of evaluations are to demonstrate that WACA improve the throughput for different source-destination pairs, and different numbers of channels and radios, in a randomly generated wireless network. Figure 1 shows the topology of the generated 25-node wireless network used for our first set of evaluations. A line between two nodes means that the link loss probability between them is less than 1. We run the evaluation 200 times. In each run, we randomly choose a pair of source and destination, which are 2-4 hops apart.
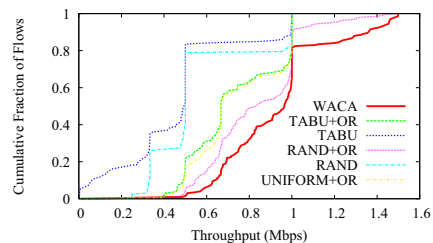
(a) 3 channels.
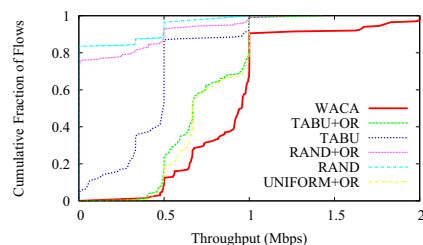
(b) 12 channels.

Fig. 2.   CDF of the throughput achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR for 200 different source-destination pairs, when there are 2 radios per node, and 3 or 12 channels.

The results show that WACA significantly improves the throughput compared with the other schemes. Figure 2 presents the cumulative distribution function (CDF) of the achieved throughputs for 200 randomly selected source-destination pairs, when there are 2 radios per node, and 3 or 12 channels. Generally, the throughput of applying EMORE is significantly higher than that of using shortest path routing. Among the schemes with EMORE, WACA performs the best in both cases. In contrast, RAND+OR's performance is not stable. It is near WACA for 3 channels, but drops dramatically when the number of channels increases from 3 to 12. This is because the connectivity can be easily guaranteed when every node uses 2 out of 3 channels, while the network is hardly connected if the nodes randomly distribute their radio onto a large range of channels. For the median case, WACA achieves 44.7%, 16.1%, and 44.5% higher throughput than TABU+OR, RAND+OR, and UNIFORM+OR for 3 channels, respectively; 42.8% and 42.6% higher throughput than TABU+OR and UNIFORM+OR for 12 channels, respectively.

Figure 2 also shows that WACA well exploits concurrent transmissions over multiple channels, and achieves much more high-throughput flows. In particular, when there are 3 channels, WACA makes 19.0% of flows having throughput more than 1 Mbps, compared with 11.5% for RAND+OR, which has the highest percentage of high throughput flows among the other schemes. When there are 12 channels, the percentage of high throughput flows (> 1 Mbps) achieved by WACA reaches

9.5%, which is much higher than percentage 0.5% got by the other schemes. The percentage of high-throughput flows drops is because 12 channels are more than enough for the 2-radio nodes, and may mislead the channel assignment algorithms to over scatter the nodes.

Furthermore, WACA eases the bottlenecks in the flows. Specifically, Figure 2(a) shows that, when there are 3 channels, 80% of the flows have a throughput higher than 0.69 Mbps, compared with the corresponding throughputs 0.50 Mbps, 0.64 Mbps, and 0.54 Mbps achieved by TABU+OR, RAND+OR, and UNIFORM+OR, respectively. A similar result is also shown for 12 channels in Figure 2(b).

Effect of Distance: To better understand the throughput improvement of WACA affected by the distance between source and destination, we category the flows by number of hops from source to destination via the shortest path.
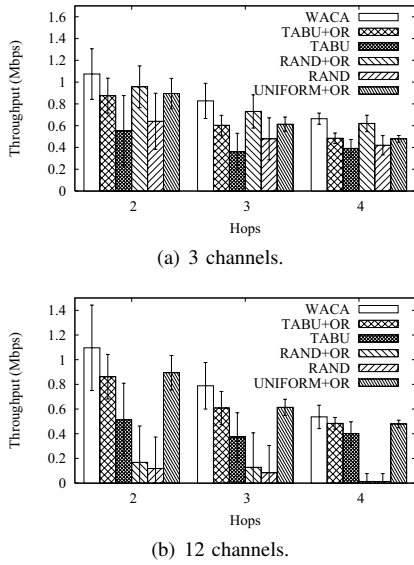

(a) 3 channels.

(b) 12 channels.

Fig. 3. Average throughput affected by the distance between source and destination, when there are 2 radios per node, and 3 or 12 channels. The distance is measured by the number of hops from source to destination via the shortest path. Standard deviations are shown using lines.

Figure 3 shows the average throughput as a function of number of hops from source to destination via the shortest path, when there are 2 radios per node, and 3 or 12 channels. Generally, the average throughput decreases with the number of hops. However, WACA always achieves the highest average throughput. Specifically, WACA achieves 6.9-13.2% and 11.9-28.5% higher throughput than the second best scheme for 3 and 12 channels, respectively (RAND+OR for 3 channels and UNIFORM+OR for 12 channels).

Effect of Number of Channels: Number of channels affects the results of the channel assignment algorithms, and thus influence the throughput achieved. To examine this factor, we randomly select 50 source-destination pairs from the network shown in Figure 1, and test the throughput of the six schemes with different number of channels available.

Figure 4 shows the average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNI-
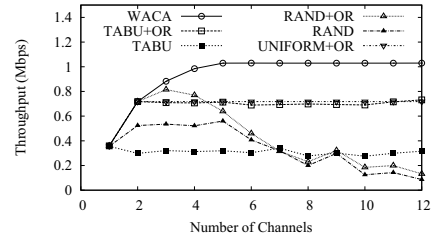

Fig. 4. Average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR, as a function of number of channels for 50 different source-destination pairs, when there are 2 radios per node.

FORM+OR, as a function of number of channels for 50 different source-destination pairs, where every node is equipped with 2 radios. WACA always performs better than the other schemes, except when the number of channels is no more than 2. The performance of TABU+OR, TABU, and UNI-FORM+OR is relatively stable when there are more than 1 channel. In contrast, the average throughputs of RAND+OR and RAND drop dramatically when the number of channels is larger than 5. This is because the connectivity of the source-destination pairs cannot be guaranteed by the random-based schemes.

Effect of Number of Radios: The number of radios equipped by each node, which determines the number possible concurrent transmissions for a node at the same time, is another important factor to affect the throughput.
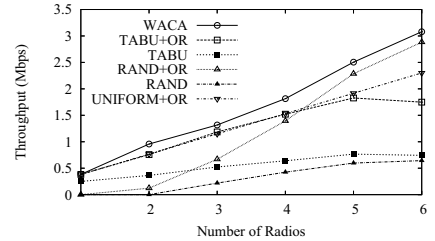

Fig. 5. Average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR, as a function of number of radios per node for 50 different source-destination pairs, when there are 12 channels.

Figure 5 shows our evaluation results on the average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR, as a function of number of radios per node for 50 different source-destination pairs, when there are 12 channels. Again, WACA always achieve the highest throughput. It is worth to note that although RAND+OR performs badly when the number of radios is small, its throughput grow dramatically, and get very close to WACA when the number of radios is large. Therefore, when the nodes have large number of radios, RAND+OR can serve as an alternative to WACA, if the nodes' computational capability is limited.

Effect of Node Density: In contrast to the first set of evaluations, which are carried out on a fixed wireless network, we change the density of nodes in a terrain area of 750 meters × 750 meters, and evaluate the end-to-end throughput from

the bottom-left node to the top-right node, in our second set of evaluations. For each density, 50 runs of evaluation are performed, with random node distribution.
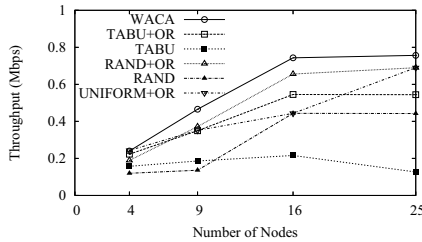

Fig. 6. Average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR, as a function of number of nodes, when there are 2 radios per node and 3 channels.

Figure 6 presents the average throughput, achieved by WACA, TABU+OR, TABU, RAND+OR, RAND, and UNIFORM+OR, affected by the number of nodes in the terrain area, when there are 2 radios per node and 3 channels. The figure shows that the throughputs are very low with poor network connectivity, *i.e.*, the number of nodes is small (4 or 9 nodes). A good network connectivity can be achieved with 16 nodes, after which adding more nodes does not help much to improve the throughput, except for UNIFORM+OR. However, WACA always achieve the highest throughput in the evaluated cases.

VI. DISCUSSION

Multiple Flows: Although, we focus on improving the throughput of a single flow in this paper, the proposed channel assignment algorithm WACA can be extended to adapt multiple flows. One of the possible ways to extend it is to compute the channel assignment for the flows sequentially. For each flow, we also take the channel assignment and workload distribution result from the previous flow as an input to the algorithm. In the initialization phase, the algorithm extends existing channel assignment to make the source and the destination connected, by assigning every free node a channel used by its neighbors. Then the algorithm iteratively update the channel assignment until the throughput cannot be improved. In the channel assignment pruning phase, only newly added assignments are allowed to be pruned. Finally, the result should also be pruned before outputting. However, there may be better ways to handle multiple flows. We will leave this problem to our future work.

Sensitivity of Parameters: The workload-aware channel assignment module employs two parameters α and β to prune the channel assignment item, on which both the duty and the workload are too small to make good contribution to the flow. Intuitively, if a node's radio does not contribute (receive or send) much on current channel, it should be tuned to other channels that are beneficiary to the flow or just left to be unused. In MORE's implementation, the parameters are set to 0.1. In our simulation, we found that when the parameters are set to 0.05, the performance WACA is good in most cases. However, α and β should be carefully picked according to the condition of the wireless mesh network.

VII. RELATED WORK

We briefly review the related works on channel assignment and opportunistic routing in this section.

A. Channel Assignment Algorithms

The channel assignment problem was first studied in cellular networks. We refer to [28] for a comprehensive survey.

A number of works were presented for wireless LANs (WLANs). For instance, Mishra et al. [29] utilized weighted graph coloring to address channel assignment for WLANs. Mishra et al. [30] used client-driven mechanisms to address the joint problem of channel assignment and load balancing in centrally managed WLANs.

Channel assignment problem was extensively studied in wireless mesh networks (WMNs). For instance, in [13], the authors have proposed channel assignment algorithms to minimize overall network interference. Another work [9], proposes a neighbor partitioning-based algorithm and a load-aware algorithm for allocating channels in multichannel networks. An interference-aware channel assignment algorithm is proposed in [31] in which the routers switch to a default channel whenever the current channel is perceived to be poor. Raniwala and Chiueh, have proposed a tree-based distributed channel assignment protocol, which considers the aggregate traffic load on a channel within the interference range [32]. Another fully distributed channel allocation protocol for multi-radio mesh networks is proposed in [33] where the objective was to maximize the utilization of the wireless spectrum over a large network while minimizing the cochannel interference. Joint channel assignment and routing algorithms for multichannel mesh networks are proposed in [8], [10], [34] with the objective of maximizing network throughput.

The channel assignment problem is also studied in other wireless networks, such as ad-hoc networks (*e.g.*, [35]) and software defined radio networks (*e.g.*, [36]).

B. Opportunistic Routing in Wireless Networks

Opportunistic routing belongs to cooperative diversity techniques (e.g. [6], [37], [38]) which take advantage of broadcast transmissions to send information through multiple concurrent relays. Nodes can combine information from multiple signals so that they can make best decisions of routing or forwarding. As an example, protocols in [37] fully exploit spatial diversity in the channel by allowing all nodes that overheard a transmission to simultaneously forward the signal. Another example is the protocol in [6], which optimizes the choice of forwarder from multiple receivers by deferring to choose each hop after transmission.

The concept of opportunistic routing was first developed by Biswas and Morris in the context of wireless mesh networks. They claimed that opportunistic routing can potentially increase the throughput and proposed an integrated routing and MAC protocol, named ExOR, to achieve the throughput gain [6]. To further improve the system throughput, Chachulski et al. designed MORE [7], which combines random network coding and opportunistic routing to avoid transmission duplication. Later, Katti et al. apply the idea of opportunistic routing down to granularity of symbol level [18]. Lin et al. [19], [20], improved the performance of opportunistic routing by

transmitting a window of multiple batches simultaneously. Rozer et al. proposed an opportunistic adaptive routing protocol SOAR [21] to support multiple simultaneous flows in wireless mesh networks. Koutsonikolas et al. [22] improved network coding based opportunistic routing protocols with a novel cumulative coded acknowledgment scheme. Laufer et al. [23] extended existing opportunistic routing protocols to better utilize wireless channels by exploiting the wireless radios' capability of working on multiple transmission bit rates specified by IEEE 802.11 protocols.

A closely related work is MCExOR [39], which is based on ExOR and employs a compressed slotted MAC layer acknowledgement mechanism. In contrast, our WACA does not rely on any MAC layer synchronization. Recently, an analysis of the end-to-end throughput bound of opportunistic routing protocol ExOR in multi-radio multi-channel wireless networks was presented by Zeng et al. in [40].

VIII. Conclusion and Future Work

In this paper, we have studied the problem of channel assignment in multi-channel, multi-radio wireless mesh networks, considering the support of opportunistic routing technique. We have presented a workload-aware channel assignment algorithm (WACA) for multi-channel opportunistic routing. Evaluation results have shown that WACA achieves significantly higher throughput than existing channel assignment algorithms. As for future work, we are interested in designing efficient joint channel assignment and opportunistic routing algorithms/protocols, that can improve the system throughput for multiple concurrent flows.

IX. Acknowledgments

References

[1] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Communications Magazine*, vol. 43, no. 9, 2005.
[2] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in *SIGCOMM'04*, Portlan, Oregon, Aug. 2004.
[3] Meraki Networks, "http://meraki.com."
[4] MuniWireless LLC, "http://www.muniwireless.com."
[5] "Ugly truth about mesh networks," www.dailywireless.org/2004/06/28/ugly-truth-about-mesh-networks.
[6] S. Biswas and R. Morris, "Opportunistic routing in multi-hop wireless networks," in *SIGCOMM'05*, Philadelphia, PA, Aug. 2005.
[7] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," in *SIGCOMM'07*, Kyoto, Japan, Aug. 2007.
[8] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *MobiCom'05*, Sep. 2005.
[9] A. Raniwala, K. Gopalan, and T. cker Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM SIGMOBILE MC2R*, vol. 8, no. 2, pp. 50–65, 2004.
[10] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *MobiCom'05*, Sep. 2005.
[11] B. Raman, "Channel allocation in 802.11-based mesh networks," in *INFOCOM'06*, Apr. 2006.
[12] A. K. Das, H. M. K. Alazemi, R. Vijayakumar, and S. Roy, "Optimization models for fixed channel assignment in wireless mesh networks with multiple radios," in *SECON'05*, Sep. 2005.
[13] P. Subramanian, H. Gupta, and S. R. Das, "Minimum interference channel assignment in multi-radio wireless mesh networks," in *SECON'06*, Sep. 2006.
[14] F. Wu, V. Raman, and N. Vaidya, "Being opportunistic or being concurrent – on designing channel assignment algorithms in multi-radio, multi-channel wireless mesh networks," in *SECON'10*, Boston, MA, Jun. 2010.
[15] W. Yue, "Analytical methods to calculate the performance of a cellular mobile radio communication system with hybrid channel assignment," *IEEE transactions on vehicular technology*, vol. 40, no. 2, pp. 453–460, 1991.
[16] D. C. Cox and D. O. Reudink, "Dynamic channel assignment in high capacity mobile communication system," *Bell System Technical Journal*, vol. 50, no. 6, pp. 1833–1857, 1971.
[17] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE JSAC*, vol. 18, no. 3, pp. 535–547, 2000.
[18] S. Katti, D. Katabi, H. Balakrishnan, and M. Medard, "Symbol-level network coding forwireless mesh networks," in *SIGCOMM'08*, SEATTLE, WA, Aug. 2008.
[19] Y. Lin, B. Li, and B. Liang, "Codeor: Opportunistic routing in wireless mesh networks with segmented network coding," in *ICNP'08*, Orlando, FL, Oct. 2008.
[20] Y. Lin, B. Liang, and B. Li, "SlideOR: oline opportunistic network coding in wireless mesh networks," in *INFOCOM'10*, Apr. 2010.
[21] E. Rozner, J. Seshadri, Y. A. Mehta, and L. Qiu, "Soar: Simple opportunistic adaptive routing protocol for wireless mesh networks," *IEEE Transactions on Mobile Computing*, vol. 8, pp. 1622–1635, 2009.
[22] D. Koutsonikolas, C.-C. Wang, and Y. C. Hu, "CCACK:efficient network coding based opportunistic routing through cumulative coded acknowledgments," in *INFOCOM'10*, Apr. 2010.
[23] R. Laufer, H. Dubois-Ferri`ere, and L. Kleinrock, "Multirate anypath routing in wireless mesh networks," in *INFOCOM'09*, Apr. 2009.
[24] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *MobiCom'03*, Sep. 2003.
[25] S. Chachulski, "Trading structure for randomness in wireless opportunistic routing," Master's thesis, MIT, May 2007.
[26] J. F. Bonnans, J. Gilbert, C. Lemaréchal, and C. Sagastizábal, *Numerical optimization, theoretical and numerical aspects, Second edition*. Springer, 2006.
[27] LINDO API, "www.lindo.com."
[28] I. Katzela and M. Naghshineh, "Channel assignment schemes for cellular mobile telecommunications: A comprehensive survey," *IEEE Personal Communications*, vol. 3(3), pp. 10–31, Jun. 1996.
[29] A. Mishra, S. Banerjee, and W. Arbaugh, "Weighted coloring based channel assignment for WLANs," *ACM SIGMOBILE MC2R*, vol. 9, no. 3, pp. 19–31, 2005.
[30] A. Mishra, V. Brik, S. Banerjee, A. Srinivasan, and W. Arbaugh, "A client-driven approach for channel management in wireless LAN," in *INFOCOM'06*, Apr. 2006.
[31] K. Ramachandran, E. Belding, K. Almeroth, and M. Buddhikot, "Interference-aware channel assignment in multi-radio wireless mesh networks," in *INFOCOM'06*, Apr. 2006.
[32] A. Raniwala and T.-C. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *INFOCOM'05*, Apr. 2005.
[33] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein, "Distributed channel assignment in multi-radio 802.11 mesh networks," in *IEEE WCNC*, March 2007.
[34] H. Wu, F. Y. K. Tan, J. Chen, Q. Zhang, and Z. Zhang, "Distributed channel assignment and routing in multiradio multichannel multihop wireless networks," *IEEE JSAC*, vol. 24, pp. 1972–1983, 2006.
[35] P. Kyasanur and N. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *ACM SIGMOBILE MC2R*, vol. 10, pp. 31–43, January 2006.
[36] Y. T. Hou, Y. Shi, and H. D. Sherali, "Optimal spectrum sharing for multi-hop software defined radio networks," in *INFOCOM'07*, May 2007.
[37] D. Laneman and G. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Transaction on Information Theory*, vol. 50, no. 12, pp. 3062 – 3080, 2004.
[38] A. K. Miu, H. Balakrishnan, and C. E. Koksal, "Improving loss resilience with multi-radio diversity in wireless networks," in *MobiCom'05*, Sep. 2005.
[39] A. Zubow, M. Kurth, and J.-P. Redlich, "Multi-channel opportunistic routing," in *Proc. of The 13th European Wireless Conference*, Paris, France, Apr. 2007.
[40] K. Zeng, Z. Yang, and W. Lou, "Opportunistic routing in multi-radio multi-channel multi-hop wireless networks," in *INFOCOM'10*, Apr. 2010.