Oblivious Routing for Wireless Mesh Networks

Jonathan Wellons and Yuan Xue
Department of Electrical Engineering and Computer Science
Vanderbilt University
Email: {jonathan.l.wellons,yuan.xue}@vanderbilt.edu

***Abstract*—Wireless mesh networks have attracted increasing attention and deployment as a high-performance and low-cost solution to last-mile broadband Internet access. Traffic routing plays a critical role in determining the performance of a wireless mesh network. To investigate the best routing solution, existing work proposes to formulate the mesh network routing problem as an optimization problem. In this problem formulation, traffic demand is usually implicitly assumed to be static and known *a priori*. Contradictorily, recent studies of wireless network traces show that the traffic demand, even being aggregated at access points, is highly dynamic and hard to estimate. Thus, in order to apply the optimization-based routing solution in practice, one must take into account the dynamic and unpredictable nature of wireless traffic demand. This paper studies the *oblivious routing* algorithm that is able to provide the optimal worst-case performance on all possible traffic demands users may impose on the wireless mesh network, where the goal is to minimize the maximum congestion appearing at all interference sets in the network over all properly scaled traffic demand patterns. To the best of our knowledge, this work is the first attempt that investigates the oblivious routing issue in the context of wireless mesh networks. A trace-driven simulation study demonstrates that our oblivious routing solution can effectively incorporate the traffic dynamics in mesh network routing.**

I. Introduction

Wireless mesh networks (*e.g.*, [1], [2]) have attracted increasing attention and deployment as a high-performance and low-cost solution to last-mile broadband Internet access. In a wireless mesh network, local access points and stationary wireless mesh routers communicate with each other and form a backbone structure which forwards the traffic between mobile clients and the Internet.

Traffic routing plays a critical role in determining the performance of a wireless mesh network. Thus it attracts extensive research recently. The proposed approaches usually fall into two ends of the spectrum. On one end of the spectrum are the heuristic routing algorithms (*e.g.*, [3]–[5]). Although many of them are adaptive to the dynamic environments of wireless networks, these algorithms lack the theoretical foundation to analyze how well the network performs globally (*e.g.*, whether the traffic shares the network in a fair fashion).

On the other end of the spectrum, there are theoretical studies that formulate mesh network routing as optimization problems (*e.g.*, [6], [7]). The routing algorithms derived from these optimization formulations can usually claim analytical properties such as resource utilization optimality and throughput fairness. In these optimization frameworks, traffic demand is usually implicitly assumed as static and known *a priori*. Contradictorily, recent studies of wireless network traces [8] show that the traffic demand, even being aggregated at access points, is highly dynamic and hard to estimate. Such observations have significantly challenged the practicability of the existing optimization-based routing solutions in wireless mesh networks.

To address this challenge and barrier to effective theoretical modeling and implementation of traffic routing to wireless mesh networks, this paper investigates the optimal routing framework which takes into account the dynamic and uncertain nature of wireless traffic demand. In particular, we will investigate how to route the traffic *obliviously*, without *a priori* knowledge of the traffic demand. Our goal is the design an *oblivious routing* algorithm that is able to provide the optimal worst-case performance on all possible traffic demands users may impose on the network.

Oblivious routing [9] is a well-studied problem for traffic engineering on the Internet. In [10], Racke et al. prove the existence of a polynomial bounded routing within a network. In [9], Azar et al. present an algorithm which solves the oblivious routing problem via an iterative linear programming (LP) formulation. Most recently, [11] has simplified the model of [9] to allow a single LP formulation. Although it is an active research topic for the Internet, to the best of our knowledge, this work is the first attempt that investigates the oblivious routing problem in the context of wireless mesh network. In fact, it is a non-trivial issue to extend the existing solutions proposed for the Internet to wireless mesh networks. The main challenge comes from the interference and channel capacity constraints which are unique to wireless networks. To address this issue, this paper uses the maximum congestion appearing at all interference sets in the network as a new routing metric and redefine the routing objective for oblivious routing. Based on the method of [11], the optimal oblivious mesh routing problem is then converted to a linear programming (LP) problem, which must be optimized over all properly scaled traffic demand patterns.

To evaluate the performance of our algorithms under a realistic wireless networking environment, we conduct trace-driven simulation study. In particular, we derive the traffic demand for the local access points of our simulated wireless mesh network based on traffic traces collected at Dartmouth College campus wireless networks. Our simulation results demonstrate that our oblivious mesh routing solution could effectively incorporate the traffic dynamics into the routing optimization of wireless mesh networks.

The original contributions of this paper are two-fold. Practically, the oblivious mesh network routing solution proposed in this paper considers traffic dynamics and uncertainty in the mesh network routing optimization. The full-fledged simulation study based on real wireless network traffic traces provides convincing validation of the practicability of this solution. Theoretically, upon the classical network congestion minimization problem for wireline networks, we redefine the concept of network congestion and extend the wireline network oblivious routing algorithm into wireless mesh networks to handle location-dependent wireless interference.

The remainder of this paper is organized as follows. Section II describes our network, interference and traffic models. Section III formulates the oblivious routing problem. Section IV presents the details of solving the oblivious routing problem. Section V presents our simulation study and results. Finally, Section VI concludes the paper.

II. Model

A. Network and Interference Model

In a multi-hop wireless mesh network, local access points aggregate and forward traffic from mobile clients which are associated with them. They communicate with each other and with the stationary wireless routers to form a multi-hop wireless backbone network. This wireless mesh *backbone* network forwards the user traffic to the gateways which are connected to the Internet. We use $w \in W$ to denote the set of gateways in the network and $s \in S$ to denote the set of local access points that generate traffic in the network. In the following discussion, local access points, gateways and mesh routers are collectively called mesh nodes and denoted by the set V (Note that $W \subset V$)[1].

In a wireless network, packet transmissions are subject to location-dependent interference. We assume that all mesh nodes have the uniform transmission range denoted by R_T. Usually the interference range is larger than its transmission range. We denote the interference range of a mesh node as $R_I = (1 + \Delta)R_T$, where $\Delta \geq 0$ is a constant. In this paper, we consider the *protocol model* presented in [12]. Let $r(u, v)$ be the distance between u and v ($u, v \in V$). In the protocol model, packet transmission from node u to v is successful, if and only if (1) the distance between these two nodes $r(u, v)$ satisfies $r(u, v) \leq R_T$; (2) any other node $w \in V$ within the interference range of the receiving node v, *i.e.*, $r(w, v) \leq R_I$, is not transmitting. If node u can transit to v directly, they form an edge $e = (u, v)$. We denote the capacity of this edge as $b(e)$ which is the maximum data rate that can be transmitted. Let E be the set of all edges. We say two edges e, e' interfere with each other, if they can not transmit simultaneously based on the protocol model. Further we define *interference set* $I(e)$ which contains the edges that interfere with edge e and e itself.

Finally, we introduce a virtual node w^* to represent the Internet. w^* is connected to each gateway with a virtual edge $e^* = (w^*, w), w \in W$. We use E' to denote the union of E and the set of all virtual edges and use V' to denote the union of V and the virtual node w^*. For simplicity, we assume that the link capacity in Internet is much larger than the wireless channel capacity, and thus the bottleneck always appears in the wireless mesh network. Under this assumption, the virtual edges could be regarded as having unlimited capacity. Note that all the virtual links do not interfere with any of the wireless transmissions.

[1]For simplicity, in this paper we assume that each node is equipped with one radio which operates on the same wireless channel as others.

B. Traffic Demand and Routing

This paper investigates the optimal routing strategy for wireless mesh *backbone* network. Thus it only considers the aggregated traffic among the mesh nodes. For ease of exposition, we only consider the aggregated traffic from gateway access points to local access points in this paper. In particular, we regard the gateway access points as the sources of all incoming traffic and the local access points, which aggregate the client traffic, as the destinations of all incoming traffic. It is worth noting that our problem formulations and algorithms could be easily extended to handle other inter-mesh-node traffic. We denote the aggregated traffic to a local access point as a *flow*. All flows will take w^* as their source. Further we denote the traffic demand from local access point $s \in S$ to w^* as d_s and use vector $\boldsymbol{d} = (d_s, s \in S)$ to denote the demand vector consisting of all flow demands.

A *routing* specifies how traffic of each flow is routed across the network. Here we assume an infinitesimally divisible flow model where the aggregated traffic flow could be routed over multiple paths and each path routes a fraction of the traffic. Thus a routing can be characterized by the fraction of each flow that is routed along each edge $e \in E'$. Formally, we use $f_s(e)$ to denote the fraction of demand from local access point s that is routed on the edge $e \in E'$. Thus, a routing could be specified by the set $\boldsymbol{f} = \{f_s(e), s \in S, e \in E'\}$. Recall that the demand of node $s \in S$ is denoted by d_s. Therefore, the amount of traffic demand from s that needs to be routed over e under routing $\boldsymbol{f}$ is $d_s f_s(e)$.

C. Schedulability

To study the mesh routing problem, we first need to understand the constraint of the flow rates. Let $\boldsymbol{y} = (y(e), e \in E)$ denote the wireless link rate vector, where $y(e)$ is the aggregated flow rate along wireless link e. Link rate vector $\boldsymbol{y}$ is said to be schedulable, if there exists a stable schedule that ensures every packet transmission with a bounded delay. Essentially, the constraint of the flow rates is defined by the schedulable region of the link rate vector $\boldsymbol{y}$.

The link rate schedulability problem has been studied in several existing works, which lead to different models [13]–[15]. In this paper, we adopt the model in [14], which is also extended in [6] for multi-radio, multi-channel mesh network. In particular, [14] presents a sufficient condition under which a link scheduling algorithm is given to achieve stability with bounded and fast approximation of an ideal schedule. [6] presents a scheme that can adjusts the flow routes and scale the flow rates to yield a feasible routing and channel assignment.

Based on these results, we have the following claim as a sufficient condition for schedulability.

Claim 1. (*Sufficient Condition of Schedulability*) The link rate vector $\boldsymbol{y}$ is schedulable *if* the following condition is satisfied:

$$\forall e \in E, \sum_{e' \in I(e)} \frac{y(e')}{b(e')} \leq 1 \quad (1)$$

III. Problem Formulation

In this section, we first investigate the formulation of optimal routing for wireless mesh backbone network under known traffic demand. Then we extend this problem formulation to the oblivious mesh network routing where the traffic demand is uncertain.

A common routing performance metric with respect to a known traffic demand is *resource utilization*. For example, link utilization is commonly used for traffic engineering in the Internet [16], whose objective is to minimize the utilization at the most congested link. However, in a multihop wireless network, such as mesh backbone network, wireless link utilization may be inappropriate as a metric of routing performance due to the location-dependent interference. On the other hand, the existing works on optimal mesh network routing [6] usually aim at maximizing the flow throughput, while satisfying the fairness constraints. In this formulation, traffic demand is reflected as the flow weight in the fairness constraints.

In light of these results, we first outline the relation between the throughput optimization problem and the congestion minimization problem, and define the utilization (so-called *congestion*) of the interference set as the routing performance metric. We further define the *performance ratio* of a routing as the ratio between its congestion and the minimum congestion under a certain demand. In order to handle uncertain traffic demand, the *performance ratio* is extended to the *oblivious performance ratio* which is the worst performance ratio a routing obtains under all possible traffic demands. The definition of *oblivious performance ratio* naturally leads to the formulation of *oblivious* mesh network routing which handles uncertain wireless network traffic.

A. Mesh Network Routing Under Known Traffic Demand

We first study the formulation of throughput optimization routing problem in a wireless mesh backbone network under known traffic demand. First we present the constraints that a routing solution needs to satisfy.

Capacity Constraint

Let $y_s(e)$ be the traffic of s that is routed over $e \in E'$. Obviously the aggregated flow rate y_e along edge $e \in E$ is given by $y_e = \sum_{s \in S} y_s(e)$. Based on the sufficient condition of schedulability in Claim 1 (Eq.(1)), we have that

$$\forall e \in E, \sum_{e' \in I(e)} \sum_{s \in S} \frac{y_s(e')}{b(e')} \leq 1 \quad (2)$$

Flow Conservation

Traffic into and out of nodes must be conserved. In particular, for the mesh routers that only relay the traffic, we have the following relations:

$$\forall u \in \{V - S\}, \forall s \in S, \sum_{e=(u,v), v \in V'} y_s(e) - \sum_{e=(v,u), v \in V'} y_s(e) = 0 \quad (3)$$

For local access points $s \in S$, let x_s be the amount of traffic (throughput) to node s, we have that

$$\forall s \in S, \sum_{e=(s,v), v \in V'} y_s(e) - \sum_{e=(v,s), v \in V'} y_s(e) = -x_s \quad (4)$$

For the virtual node w^* which represents the Internet that originates all the traffic, we have

$$\forall s \in S, \sum_{e=(w^*,v), v \in V'} y_s(e) - \sum_{e=(v,w^*), v \in V'} y_s(e) = \sum_{s \in S} x_s \quad (5)$$

Recall that d_s is the demand of local access point s. Consider the fairness constraint that, for each flow of s, its throughput x_s being routed is in proportion to its demand d_s. Our goal is to maximize λ (so called *scaling factor*) where at least $\lambda \cdot d_f$ amount of throughput can be routed for node s. Summarizing the above discussions, the throughput optimization routing with fairness constraint is then formulated as the following linear programming (LP) problem.

$$\begin{aligned}
&\mathbf{P_T}: && (6)\\
&\text{maximize} \quad \lambda && (7)\\
&\text{subject to} \quad \sum_{e' \in I(e)} \sum_{s \in S} \frac{y_s(e)}{b(e')} \leq 1, \forall e \in E && (8)\\
&\sum_{e=(u,v)} y_s(e) - \sum_{e=(v,u)} y_s(e) = 0, && (9)\\
&\forall u \in \{V - S\}, \forall v \in V', \forall s \in S &&\\
&\sum_{e=(s,v)} y_s(e) - \sum_{e=(v,s)} y_s(e) = -\lambda \cdot d_s, && (10)\\
&\forall v \in V', \forall s \in S &&\\
&\sum_{e=(w^*,v)} y_s(e) - \sum_{e=(v,w^*)} y_s(e) = \lambda \sum_{s \in S} d_s && (11)\\
&\forall v \in V' &&\\
&\lambda \geq 0, \forall s \in S, \forall e \in E, y_s(e) \geq 0, && (12)
\end{aligned}$$

Note that the above problem formulation follows the classical maximum concurrent flow problem. Although being extensively used to study mesh network routing schemes under known and fixed traffic demand [6], [17], such throughput optimization problem formulation is hard to extend to handle the case of uncertain demand.

In light of this need, we proceed to study the congestion minimization routing. This differs from the throughput optimization problem where the traffic demand may not be completely routed subject to the constraints of the network capacity. Rather, the congestion minimization problem will route all the traffic demands which may violate the network capacity constraint, and thus the goal is to minimize the network congestion.

Let $y'_s(e)$ be the traffic of s on edge e under traffic demand d_s.

$$y'_s(e) = f_s(e) \cdot d_s \tag{13}$$

Formally, we define the *congestion* of an interference set $I(e)$ using its utilization (*i.e.*, the ratio between its traffic load and the channel capacity) and denote it as $\rho(e)$:

$$\rho(e) = \sum_{e' \in I(e)} \sum_{s \in S} \frac{y'_s(e)}{b(e)} = \sum_{e' \in I(e)} \sum_{s \in S} \frac{f_s(e) \cdot d_s}{b(e)} \tag{14}$$

Further, we define the *network congestion* $\rho = \max_{e \in E} \rho(e)$ as the maximum congestion among all the interference sets $I(e)$. The congestion minimization routing problem is then formulated as follows:

$$\mathbf{P_C}: \tag{15}$$
$$\text{minimize} \quad \rho \tag{16}$$
$$\text{subject to} \quad \sum_{e' \in I(e)} \sum_{s \in S} \frac{y'_s(e)}{b(e')} \le \rho, \forall e \in E \tag{17}$$
$$\sum_{e=(u,v)} y'_s(e) - \sum_{e=(v,u)} y'_s(e) = 0, \tag{18}$$
$$\forall u \in \{V - S\}, \forall v \in V', \forall s \in S$$
$$\sum_{e=(s,v)} y'_s(e) - \sum_{e=(v,s)} y'_s(e) = -d_s, \tag{19}$$
$$\forall v \in V', \forall s \in S$$
$$\sum_{e=(w^*,v)} y'_s(e) - \sum_{e=(v,w^*)} y'_s(e) = \sum_{s \in S} d_s \tag{20}$$
$$\forall v \in V'$$
$$\forall s \in S, \forall e \in E, y'_s(e) = f_s(e) \cdot d_s \ge 0 \tag{21}$$
$$\rho \ge 0, \tag{22}$$

To reveal the relation between $\mathbf{P_T}$ and $\mathbf{P_C}$, we let $\rho = \frac{1}{\lambda}$ and $y'_s(e) = \frac{y_s(e)}{\lambda}$. Problem $\mathbf{P_C}$ is then transformed equivalent to the throughput optimization problem $\mathbf{P_T}$.

B. Oblivious Mesh Network Routing

Extensive research has been conducted on the optimal mesh network routing problem formulated in Section III-A. The results from these studies are thus based on the assumption of fixed and known traffic demand. Recent studies [8], however, show that the traffic demand, even being aggregated at access points, is highly dynamic and hard to estimate. To address this issue, in this paper, we study the routing solutions that are robust to the changing traffic demands.

First we need to study the performance metric that could characterize a "good" routing solution. Based on the discussions in Section III-A, we start with the network congestion $\rho(\boldsymbol{f}, \boldsymbol{d})$ under a certain routing $\boldsymbol{f}$ and traffic demand vector $\boldsymbol{d}$, *i.e.*, $\rho(\boldsymbol{f}, \boldsymbol{d}) = \max_{e \in E} \sum_{e' \in I(e)} \sum_{s \in S} \frac{y'_s(e)}{b(e)}$. An *optimal routing* $\boldsymbol{f}^{opt}(\boldsymbol{d})$ for a certain demand vector $\boldsymbol{d}$ would give the minimum congestion, *i.e.*,

$$\rho^{opt}(\boldsymbol{d}) = \min_{\boldsymbol{f}} \rho(\boldsymbol{f}, \boldsymbol{d}) \tag{23}$$

Now we define the *performance ratio* $\gamma(\boldsymbol{f}, \boldsymbol{d})$ of a given routing $\boldsymbol{f}$ on a given demand vector $\boldsymbol{d}$ as the ratio between the network congestion under $\boldsymbol{f}$ and the minimum congestion under the optimal routing, *i.e.*,

$$\gamma(\boldsymbol{f}, \boldsymbol{d}) = \frac{\rho(\boldsymbol{f}, \boldsymbol{d})}{\rho^{opt}(\boldsymbol{d})} \tag{24}$$

Performance ratio γ measures how far $\boldsymbol{f}$ is from being optimal on the demand $\boldsymbol{d}$. Now we extend the definition of performance ratio to handle uncertain traffic demand. Let $\boldsymbol{D}$ be a set of traffic demand vectors. Then the performance ratio of a routing $\boldsymbol{f}$ on $\boldsymbol{D}$ is defined as

$$\gamma(\boldsymbol{f}, \boldsymbol{D}) = \max_{\boldsymbol{d} \in \boldsymbol{D}} \gamma(\boldsymbol{f}, \boldsymbol{d}) \tag{25}$$

A routing $\boldsymbol{f}^{opt}$ is optimal for the traffic demand set $\boldsymbol{D}$ if and only if

$$\boldsymbol{f}^{opt} = \arg\min_{\boldsymbol{f}} \gamma(\boldsymbol{f}, \boldsymbol{D}) \tag{26}$$

When the set $\boldsymbol{D}$ includes all possible demand vectors $\boldsymbol{d}$, we refer to the performance ratio as the *oblivious performance ratio*. The oblivious performance ratio is the worst performance ratio a routing obtains with respect to all possible demand vectors. To study the optimal routing strategy under uncertain traffic demand, we are interested in the *optimal oblivious routing* problem which finds the routing that minimizes the *oblivious performance ratio*. We call this minimum value the *optimal oblivious performance ratio*.

It is worth noting that the performance ratio γ is invariant to scaling. Thus to simplify the problem, we only consider traffic demand vectors $\boldsymbol{D}$ that satisfies $\rho^{opt}(\boldsymbol{d}) = 1$, instead of considering all possible traffic vectors. In this case,

$$\gamma(\boldsymbol{f}, \boldsymbol{D}) = \max_{\boldsymbol{d} \in \boldsymbol{D}} \rho(\boldsymbol{f}, \boldsymbol{d}) \tag{27}$$

Formally, the *optimal oblivious routing* problem for wireless mesh network is given as follows.

$$
\begin{aligned}
&\mathbf{P_O}: && (28)\\
&\text{minimize} \quad \rho && (29)\\
&\text{subject to} \quad \sum_{e' \in I(e)} \sum_{s \in S} \frac{y'_s(e)}{b(e')} \leq \rho, \forall e \in E && (30)\\
&\sum_{e=(u,v)} y'_s(e) - \sum_{e=(v,u)} y'_s(e) = 0, && (31)\\
&\forall u \in \{V - S\}, \forall v \in V', \forall s \in S &&\\
&\sum_{e=(s,v)} y'_s(e) - \sum_{e=(v,s)} y'_s(e) = -d_s, && (32)\\
&\forall v \in V', \forall s \in S &&\\
&\sum_{e=(w^*,v)} y'_s(e) - \sum_{e=(v,w^*)} y'_s(e) = \sum_{s \in S} d_s && (33)\\
&\forall v \in V' &&\\
&\forall s \in S, \forall e \in E, y'_s(e) = f_s(e) \cdot d_s \geq 0 && (34)\\
&\rho \geq 0, \forall \boldsymbol{d} \text{ with } \rho^{opt}(\boldsymbol{d}) = 1 && (35)
\end{aligned}
$$

IV. Algorithm

The oblivious mesh routing problem $\mathbf{P_O}$ cannot be solved directly, because it is taken over all demand vectors, and $\rho^{opt}(\boldsymbol{d})$ is an embedded maximization in the minimization problem.

In [9], a polynomial-time method is given to solve an non-linear programming problem over all possible demand matrices using an ellipsoid method and separation oracle. Though theoretically sound, this method is hard to be implemented for practical use.

Here we follow the same idea as presented by Applegate and Cohen in [11]. This method provides a LP formulation of the oblivious routing problem. The key insight is to look at the dual problem of the slave LPs of the original oblivious routing problem. To adopt this method, we introduce interference set weights $\pi_e(e')$ in the dual formulation for every pair of interference sets e, e'. Further let $p_e(s)$ correspond to the length of the shortest path between local access point s and virtual gateway w^*. $\mathbf{D_O}$ summarizes the LP formulation of oblivious mesh routing based on the dual formulation of its slave LPs.

It is worth noting that this set of equations in $\mathbf{D_O}$ represents a linear programming problem, thus we can solve it directly with a LP solver. Our choice of LP solver was *lp_solve* [18], an open source Mixed Integer Linear Programming (MILP) solver.

$$
\begin{aligned}
&\mathbf{D_O}:\\
&\text{minimize} \quad \rho\\
&\forall e, e' \in E : \sum_e b(e)\pi_e(e') \leq \rho\\
&\forall e \in E, \forall s \in S :\\
&\qquad \sum_{e' \in I(e)} f_s(e')/b(e') \leq p_e(s)\\
&\forall e \in E, \forall s \in S, \forall e' = s' \to w^* :\\
&\qquad \pi_e(e') + p_e(s) - p_e(s') \geq 0\\
&\forall e, e' \in E, \pi_e(e') \geq 0\\
&\forall e \in E, \forall s \in S : p_e(s) \geq 0
\end{aligned}
$$

V. Simulation Study

A. Simulation Setup

We evaluate the performance of our algorithm with a simulation study. In the simulated wireless mesh network, 60 mesh nodes were randomly deployed over a $1000 \times 2000m^2$ region. The simulated network topology is shown in Fig. 1. 10 nodes at the edge of this network are selected as the local access points (LAP) that forward traffic for clients. 2, 4 and 8 nodes near the center of the deployed region were selected as gateway access points as shown in Fig. 1. We have evaluated the performance of the algorithm with each of the three sets of gateways chosen. Each mesh node has a transmission range of $250m$ and an interference range of $500m$. The data bit rate $b(e)$ is set as 54 Mbps for all $e \in E$.

B. Traffic Demand Generation

To realistically simulate the traffic demand at each LAP, we employ traces collected in a campus wireless LAN network. The network traces used in this work were collected in Spring 2002 at Dartmouth College and provided by CRAWDAD [19]. By analyzing the *snmp* log trace at each access point, we are able to derive their incoming and outgoing traffic volume beginning 12:00AM, March 25, 2002 EST. We argue that the LAPs of a wireless mesh network serve a similar role as the access points of wireless LAN networks at aggregating and forwarding client traffic. Thus, we select the access points from the Dartmouth campus wireless LAN and assign their traffic traces to the LAPs in our simulation. The traffic assignment is given in Table I.

We evaluate and compare different traffic routing strategies for this simulated network. In addition to Oblivious Routing (OBR), we consider the Oracle Routing strategy and Shortest Path Routing.

- *Oracle Routing (OR)*. The traffic demand is known *a priori*. It runs a straightforward algorithm based on this demand. This routing solution is rerun every hour based on the up-to-date traffic demand from the trace and returns the optimal set of routes. As a result, no other routing algorithm can outperform OR, and we used it as

AP	31AP3	34AP5	55AP4	57AP2	62AP3	62AP4	82AP4	94AP1	94AP3	94AP8
Node ID	22	18	57	5	55	20	53	3	56	27

TABLE I
MAPPING OF TRACE DATA LAPS TO SIMULATION LAPS

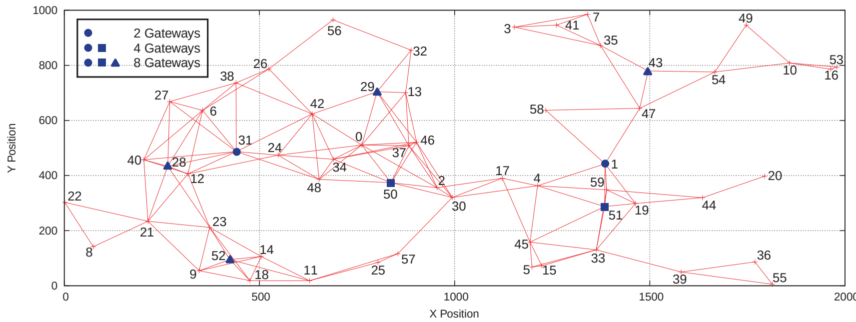

Fig. 1. Mesh Network Topology.

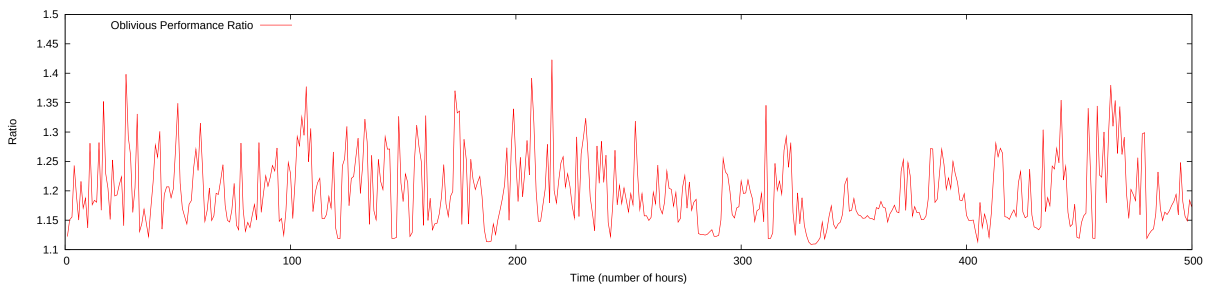

Fig. 2. Oblivious Performance Ratio Over Time, 4 Gateways

a baseline. In the figures in this section, we represent the quality of the network's oblivious and shortest path routings as a function of the demands by their ratio with respect to OR.

- *Shortest-Path Routing (SPR)*. This strategy is agnostic of traffic demand, and returns fixed routing solution purely based on the shortest distance (number of hops) from each mesh node to the gateway. The purpose to evaluate this strategy is to quantitatively contrast the advantage of our traffic-predictive routing strategies.

C. Simulation Results

First we simulate the Oblivious Routing (OBR), Oracle Routing (OR), and Shortest-Path Routing (SPR) strategies respectively over the network configuration with 4 gateways. In Fig. 2, the performance ratio of Oblivious Routing and Oracle Routing ($ratio(\gamma) = \frac{\rho_{ORB}}{\rho_{OR}}$) is plotted for each hour since the beginning of the trace collection. The ratio generally remains in the range of [1.15, 1.3], with occasional spikes. This result shows that our oblivious routing strategy performs competitively against the oracle routing strategy even without the knowledge of traffic demand.

We compare the performance ratio of Oblivious Routing and Oracle Routing ($ratio_{ORB} = \frac{\rho_{ORB}}{\rho_{OR}}$) and the performance ratio of Shortest Path Routing and Oracle Routing ($ratio_{SPR} = \frac{\rho_{SPR}}{\rho_{OR}}$) over an arbitrary chosen block of one hundred hours in Fig. 3. From the figure, we observe that although both algorithms are intermittently superior, oblivious routing outperforms SPR in most of the time. This observation is illustrated directly in Fig. 4, which shows the sorted performance ratios ($ratio_{ORB}, ratio_{SPR}$). The figure shows that the shortest path routing performs better in cases where very little congestion occurs, but for the majority of cases, oblivious routing is substantially better.

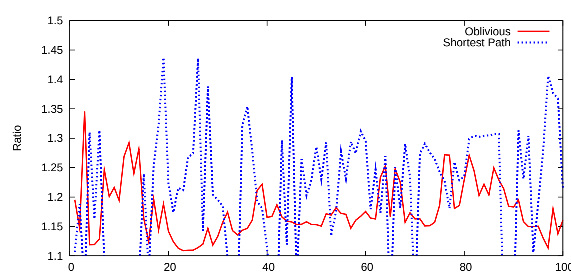

Fig. 3. Comparison of Oblivious Routing and Shortest Path Routing Over Time, 4 Gateways

Next we proceed to examine the distribution of congestion appearing at all the interference sets in our topology at an arbitrary but typical congested hour, which is plotted in Fig. 5. The figure shows that several sets reach their fully

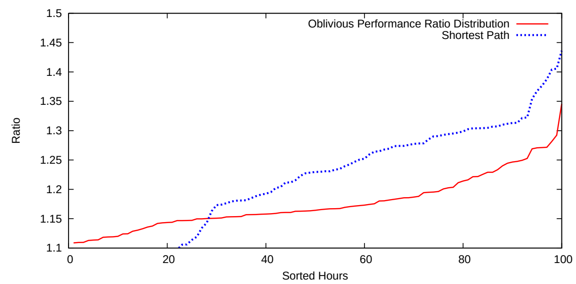

Fig. 4. Sorted Oblivious Performance Ratio Comparison, 4 Gateways

congested peak at the same time. This can be explained by the LP formulation which attempts to minimize the maximal congestion and prevent any single interference set (region) from being too congested. In addition, we could observe that the traffic is well balanced across different interference sets in the network.

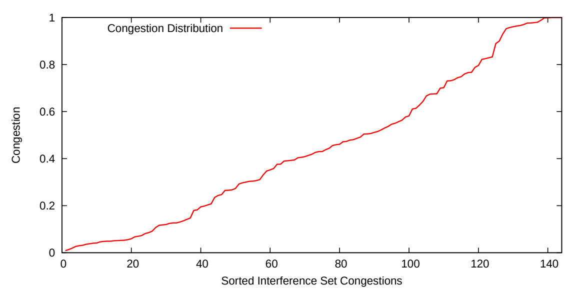

Fig. 5. Sorted Congestion over All Interference Sets, 4 Gateways

In order to better understand the relation between the number of gateways and the oblivious performance ratio, the simulation was also run with 2 and 8 gateways. Fig. 6 shows the sorted oblivious performance ratios in these three cases.

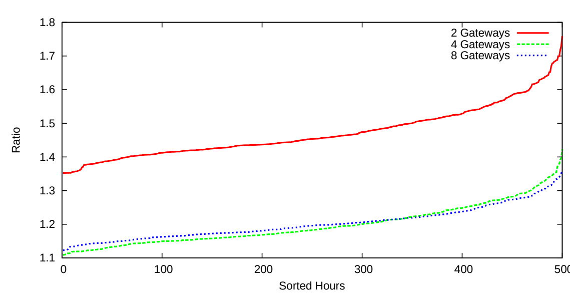

Fig. 6. Sorted Oblivious Performance Ratio, 2, 4, and 8 Gateways

The figure shows that the oblivious performance ratio tends to be higher with 2 gateways, presumably because this case requires longer paths with more potential bottlenecks during unfavorable demands. The figure also shows that 8 gateways provides approximately the same performance as 4 gateways as the routing efficiency advantages of additional gateways begins to plateau. Perhaps a larger network would better distinguish the usefulness of more gateways.

VI. Concluding Remarks

This paper studies the oblivious routing strategies for wireless mesh networks. Different from existing works which implicitly assume traffic demand as static and known *a priori*, this work considers the traffic demand uncertainty. By defining the routing objectives based on the maximum congestion over all interference sets for all possible traffic demands, we formulate the oblivious mesh network routing problem and convert it into a linear programming problem which could be easily solved via any LP solver. Simulation study is conducted based on the traffic demand from the real wireless network traces. The results show that our oblivious mesh network routing solution could effectively incorporate the traffic demand dynamics and uncertainty and perform competitively against the optimal (oracle) routing which knows the traffic demand *a priori*.

References

[1] "Seattle wireless," http://www.seattlewireless.net.
[2] "Mit roofnet," http://www.pdos.lcs.mit.edu/roofnet/.
[3] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of ACM Mobicom*, 2004.
[4] S. Biswas and R. Morris, "Exor: opportunistic multi-hop routing for wireless networks," in *Proc. of ACM SIGCOMM*, 2005.
[5] A. Raniwala and T. Chiueh, "Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network," in *Proc. of IEEE INFOCOM*, 2005.
[6] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. of ACM MobiCom*, 2005.
[7] J. Tang, G. Xue, and W. Zhang, "Maximum throughput and fair bandwidth allocation in multi-channel wireless mesh networks," in *Proc. of IEEE INFOCOM*, 2006.
[8] X. Meng, S. H. Y. Wong, Y. Yuan, and S. Lu, "Characterizing flows in large wireless data networks," in *Proc. of ACM MobiCom*, 2004.
[9] Yossi Azar, Edith Cohen, Amos Fiat, Haim Kaplan, and Harald Racke, "Optimal oblivious routing in polynomial time," *J. Comput. Syst. Sci.*, vol. 69, no. 3, 2004.
[10] Harald Racke, "Minimizing Congestion in General Networks," in *FOCS '02: Proc. of the 43rd Symposium on Foundations of Computer Science*, Washington, DC, USA, 2002, pp. 43–52, IEEE Computer Society.
[11] David Applegate and Edith Cohen, "Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs," in *Proc. of ACM SIGCOMM*, 2003, pp. 313–324.
[12] P. Gupta and P.R. Kumar, "The capacity of wireless networks," *IEEE Trans. on Information Theory*, pp. 388–404, 2000.
[13] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact on interference on multi-hop wireless network performance," in *Proc. of Mobicom*, September 2003.
[14] V. S. Anil Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan, "Algorithmic aspects of capacity in wireless networks," in *Proc. of ACM SIGMETRICS*, 2005, pp. 133–144.
[15] Yuan Xue, Baochun Li, and Klara Nahrstedt, "Optimal resource allocation in wireless ad hoc networks: A price-based approach," *IEEE Transactions on Mobile Computing*, vol. 5, no. 4, pp. 347–364, April 2006.
[16] H. Wang, H. Xie, L. Qiu, Y. R. Yang, Y. Zhang, and A. Greenberg, "Cope: traffic engineering in dynamic networks," in *Proc. of ACM SIGCOMM*, 2006.
[17] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. of ACM Mobicom*, 2005.
[18] "The lp_solve mixed integer linear programming (milp) solver," http://lpsolve.sourceforge.net/5.5/.
[19] "A community resource for archiving wireless data at dartmouth," http://crawdad.cs.dartmouth.edu/.