Robust Routing and Scheduling in Wireless Mesh Networks

Wei Wang
Department of Computer Science
University of California, Davis
Davis, CA 95616
Email: weiwang@ucdavis.edu

Xin Liu
Department of Computer Science
University of California, Davis
Davis, CA 95616
Email: liu@cs.ucdavis.edu

Dilip Krishnaswamy
Adv Tech R&D, Chief Scientist Office
Qualcomm Corporation
San Diego, CA 92130
Email: dilip@ieee.org

Abstract— **Joint routing-and-scheduling has been considered in wireless mesh networks for its significant performance improvement. While existing work assumes it, accurate traffic information is usually not available due to traffic dynamics, as well as inaccuracy and delay in its measurement and dissemination. In addition, the joint routing and scheduling usually requires a centralized controller to calculate the optimal routing and scheduling and distribute such policies to all the nodes. Thus, even if the accurate traffic information is always available, the central controller has to compute the routing and scheduling repeatedly because the traffic demands change continuously. This leads to prohibitive computation and distribution overhead. Therefore, in this paper, we propose a joint routing-scheduling scheme that achieves robust performance under traffic information uncertainty. In particular, it achieves worst-case optimal performance under a range of traffic conditions. This unique feature validates the use of centralized routing and scheduling in wireless mesh networks. As long as the traffic variation is within the estimation range, the routing and scheduling do not need to be recomputed and redistributed. Our simulation shows that the proposed scheme achieves very good average performance under a large error margin on traffic estimation and is robust when the estimation largely deviates from the actual traffic patterns. Our scheme provides insights on the desired properties of multipath routing, namely, spatial reuse and load balancing.**

I. Introduction

There has been a significant amount of interest in wireless mesh networking for its deployment flexibility and low cost. Joint routing and (TDM-based) scheduling has been considered in wireless mesh networks to improve performance [1], [2], [3], [4]. Most existing work in joint routing and scheduling assumes accurate traffic information, e.g., active source-destination (s-d) pairs and the traffic demand for each pair. However, accurate traffic information is hardly available because traffic is dynamic and its accurate measurement is rarely available. In addition, dissemination of traffic information incurs delay and signaling overhead.

In this paper, we propose a robust routing-and-scheduling scheme to handle traffic dynamics and traffic information uncertainty. To elaborate, the proposed scheme optimizes the worst-case performance (in terms of relative congestion level) for a range of traffic conditions and it performs under a whole spectrum of traffic information granularity. On one end of the spectrum where prefect traffic information is available, the proposed scheme achieves the same performance as other optimal routing and scheduling schemes suggested in the literature. On the other end of the spectrum where there is no traffic information available, the proposed scheme achieves robust performance while other schemes cannot generate desired routing and schedule for the lack of information. In general, the amount of traffic information available is between the two extremes. For example, we may know that the traffic pattern is "single-sink", i.e., all nodes communicate with a common sink node (e.g., a portal or gateway in wireless mesh networks). In addition, the traffic demand between an s-d pair can be estimated to be within a certain range (e.g., 2.0Mbps-2.6Mbps) [1]. Such traffic information can be utilized in the proposed scheme. The more accurate the information, the better the performance of the proposed scheme. In addition, the scheme is robust even when the estimate deviates from the reality.

Routing has been extensively studied in multi-hop wireless networks [5], [6]. Most of the routing protocols are shortest path routing protocols, although the path

[1] In this paper, we use "traffic pattern" to denote the active source-destination pairs, and "traffic demand" to denote the actual throughput requirement for a particular source-destination pair.

length can be defined using metrics other than the number of hops or the actual distance [7]. Multipath routing has also been studied for fault tolerance and performance improvement [8], [9]. Routing only without the consideration of the interference can lead to poor performance in wireless networks. For example, [10] shows that in a random topology with heavy traffic load, shortest path routing leads to heavy congestion, and correspondingly significant interference in the center of the network.

Therefore, joint routing and scheduling has been studied to achieve optimal performance under various scenarios. In [1], interference constraint is added into the multi-commodity flow formulation of the network to form a single linear program (LP). The active s-d pairs are required information for the formulation. It is implied that the actual demand for each s-d pair is saturated. However, traffic demand can be arbitrary in practice. Different demand vector may lead to different optimal routing and scheduling. Cruz et al. [2] considers joint routing, scheduling and power control. They assume every link in the network is active and thus do not consider end-to-end flows. In [4], a joint channel allocation, routing, and scheduling scheme is proposed for multi-channel multi-radio wireless networks. This formulation also requires similar information as in [1]. There are two problems which affect the practical deployment of the existing centralized routing and scheduling approaches. First, optimal routing and scheduling schemes assume accurate traffic information, which is often not available. Second, joint routing and scheduling optimization requires a central controller to collect the topology and traffic information of the whole network, compute the optimal routing and scheduling policies, and then distribute the policies to each individual node. Thus, even if the accurate traffic information is always available, the optimal policies have to be repeatedly computed and distributed since traffic demands vary over time. Although the centralized routing and scheduling approach can greatly improve the throughput performance of the network, such prohibitive *overhead* prevents it from being deployed.

Researchers have made great progress in traffic-oblivious routing. A traffic-oblivious routing scheme requires limited information on traffic pattern and achieves worst-case optimal performance under the given range of traffic patterns. A recent seminal work of R*ä*cke [11] shows that for any network there is an oblivious routing with a polylog competitive ratio with respect to congestion. In a follow-up work [12], the authors show how to compute the oblivious ratio in polynomial time. However, their formulation consists of polynomial number of variables, but infinite number of constraints. In [13], the authors further reduce the complexity to compute the oblivious ratio of a given network through a single LP with polynomial number of variables and constraints. In [14], the authors apply similar approaches to derive an energy-efficient traffic-oblivious routing scheme for wireless sensor networks. Our work is motivated by previous research. In the routing-only papers [11], [12], [13], [14], the main challenge is to handle the infinite number of possible traffic demands for optimal worst-case performance. Our contribution is to formulate a joint routing-and-scheduling scheme that takes into account of interference among links, which is critical and unique in a wireless network. The existing algorithm does not imply ours, as explained in more detail in Section III-A.

In this paper, we propose a robust routing-and-scheduling scheme under traffic dynamics and traffic information uncertainty. We show that the proposed scheme achieves optimal worst-case performance, as well as good average performance under traffic dynamics. It does not have the aforementioned two implementation problems, namely information availability and prohibitive overhead. First, it does not require accurate traffic information. Specifically, it only needs a rough estimation of the traffic demand, which can range from accurate information to no information at all. Second, the routing and scheduling policies need not be recomputed and redistributed often. As long as the actual traffic is within the estimation range, no update is needed. Therefore, such a scheme is feasible and with affordable overhead. In practice, centralized structure has also been considered in wireless mesh networks to provide better quality of service, especially for enterprise applications. For example, Cisco's unified wireless network architecture enables centralized control and include additional control and management functionalities [15]. Our scheme can be applied in such architecture to further improve the system performance.

We consider the possibility to exploit the oblivious routing with the decentralized MAC in case the centralized scheduling is hard to implement. Note that, it is different from generating the oblivious routing only as in the previous work. We actually generate the joint routing and scheduling. But we only use the routing with the 802.11 MAC. We study in detail the output from the scheme to understand the desirable properties of routing and scheduling that is robust to traffic variations. The routing generated in our framework is multi-path routing that *exploits spatial reuse* in the network and *provides tolerance to traffic dynamics.* Its tolerance to traffic dynamics is due to the load balancing effect of splitting flows to multiple paths. This is important in wireless networks because the load on a link interferes with its neighboring links. Therefore, good multi-path routing is important to achieve robust performance in a wireless network. Routes generated by our work provide insights to the design of decentralized routing and medium access schemes (i.e., sensing-based or distributed TDM schemes).

The rest of the paper is organized as follows. In Section II, we introduce the system model and define routing and schedulability formally. In Section III, we present our traffic-oblivious routing and scheduling (TORS) scheme. We consider two cases: without traffic information and with some traffic information. Simulation results are presented in Section IV. Conclusions are in Section V.

II. System Model

A. Interference

We consider the following interference model. Let l_{ij} be the link between nodes i and j. Links l_{ij} and l_{uv} interfere with each other if either of their transmissions can cause interference at the other. More precisely,

$$\begin{aligned} & l_{ij} \text{ and } l_{uv} \text{ interfere with each other} \\ & \text{if and only if} \\ & \quad i=u \text{ or } i=v \text{ or } j=u \text{ or } j=v \\ & \text{or at least one of the four links exist:} \\ & \quad l_{iu}, l_{iv}, l_{ju} \text{ and } l_{jv} \end{aligned} \tag{1}$$

This is a widely used interference model in the research community, e.g., [16], [17], because it models the interference in the IEEE 802.11-based networks where the receiver also sends back the acknowledgment frame during the data transfer. Therefore, both the sender and the receiver may cause interference at other nodes. Such interference relationship can be measured statically when the network is setup. We note that in practice, the interference among links rely on many factors, such as the transmission power, link quality, reception threshold, etc. A static interference model reflects the interference relationship among links statistically. Furthermore, in the performance evaluations, we validate this interference model using packet-level simulations where the practical physical layer is modeled (i.e., the packet can be correctly received only when the received SINR is larger than the reception threshold.).

We use conflict graph to model interference among links, as discussed in [1]. In the conflict graph, each vertex is a link in the original communication connectivity graph. Two vertices are connected with an edge in the conflict graph if the corresponding communication links cannot transmit simultaneously. Therefore, the conflict graph captures mutual interference among links. Due to space limit, please refer to Fig. 4 and Table 1 in [1] for an example of conflict graph. Links are scheduled based on maximal independent sets. In graph theory, an independent (vertex) set is a subset of vertices such that no two vertices in the subset represent an edge in the graph. A maximal independent set is therefore an independent set containing the largest possible number of vertices. The maximal independent sets are derived from the conflict graph. The vertices (the corresponding communication links in the connectivity graph) within a maximal independent set can transmit simultaneously without collision. Let I_i, $i = 1, \cdots, K$, be the maximal independent sets. The schedulability constraint can be formulated as:

$$\begin{aligned} & \sum_{i=1}^{K} \lambda_i \leq 1 \\ & g(l) \leq \sum_{l \in I_i} \lambda_i cap(l), \end{aligned} \tag{2}$$

where λ_i is the portion of time allocated to independent set I_i, $cap(l)$ is the capacity of link l, and $g(l)$ is the load on link l. A link can belong to multiple independent sets. Therefore, $\sum_{l \in I_i} \lambda_i$ is the total portion of time that link l can transmit. We denote $\lambda = \{\lambda_i\}$ the scheduling.

Finding all maximal independent sets is NP-complete. Thus, we exploit the technique in [1] to find a subset of them within a given amount of *effort*. The effort needed to find the maximal independent sets is controllable and independent of the network size. Generally, the more the effort spent, the more accurate the result will be. In our simulations, we set the number of efforts to be sufficiently large such that all maximal independent sets are found in general. Fortunately, this is a one-time overhead. As long as the topology is not changed, the maximal independent sets will not be recomputed.

We note that various interference models have been proposed in the literature, including graph-based, SINR-based, and measurement-based models. It is important to note that *none of our algorithms relies on the specifics of the interference model used.* The only input from the interference model required by our algorithm is the sets of links that can be scheduled simultaneously. Different interference models may result in different sets of links that can transmit simultaneously. As long as such an input is provided by an interference model, the proposed algorithm can be applied without modifications.

B. Traffic Patterns

We use a *traffic matrix* (TM) to represent the traffic pattern and demand. A traffic matrix is an $n \times n$ matrix where the diagonal entries are set to 0. A traffic matrix provides the traffic demand of each s-d pair. The ith row and jth column of the traffic matrix, d_{ij}, denotes the amount of traffic from the source node i to the destination node j. We use the traffic matrix to model different amount of traffic information available. We consider three cases: 1) no information is available; i.e., $0 \leq d_{ij} \leq \infty$ for all i, j; 2) known s-d pairs but no information on the corresponding traffic demand; i.e., $0 \leq d_{ij} \leq \infty$ for s-d pairs and $d_{ij} = 0$ for non-s-d pairs; and 3) known s-d pairs with a range of traffic demand; i.e., $a_{ij} \leq d_{ij} \leq b_{ij}$ for s-d pairs and $d_{ij} = 0$ for non-s-d pairs.

C. Routing

A routing specifies how traffic of each s-d pair is routed across the network. We denote $f = \{f_{ij}(e)\}$ a routing, where $f_{ij}(e)$ specifies the *proportion* of demand from i to j that is routed on the edge e. When f is used to route a demand d_{ij}, the load of this demand on edge e is $d_{ij} f_{ij}(e)$. Thus, a routing f is formally defined as

$$\forall \text{ edges } e, \forall \text{ nodes } i, j \neq i : f_{ij}(e) \geq 0 \quad (3)$$

$$\forall \text{ nodes } i, j \neq i, k \neq i, j : \sum_{e \in \text{out}(k)} f_{ij}(e) - \sum_{e \in \text{in}(k)} f_{ij}(e) = 0 \quad (4)$$

$$\forall \text{ nodes } i, j \neq i : \sum_{e \in \text{out}(i)} f_{ij}(e) - \sum_{e \in \text{in}(i)} f_{ij}(e) = 1, \quad (5)$$

where in(k) and out(k) denote the set of edges pointing to and from node k respectively, (4) is the flow conservation constraint at the intermediate node, i.e., if a node is neither the source nor the destination of the flow, then the amount of flow "in" equals the amount of flow "out", and (5) is the flow conservation constraint at the source node. Note that the flow conservation constraint on the destination node is redundant given (4) and (5). We should also note that arbitrary multi-path routing is considered in f by definition.

III. Traffic-Oblivious Routing and Scheduling

Motivated by existing work on traffic-oblivious routing in Internet, we propose a traffic-oblivious routing-and-scheduling (TORS) scheme that captures the interference constraints in wireless networks. The objective of a traffic-oblivious scheme is to handle uncertainty in traffic information and achieve worst-case optimal performance under the given range of traffic information. We first introduce the notations.

The performance metric of a given routing-and-scheduling pair, (f, λ), with respect to a TM D, is the maximum congestion level among all links via (f,λ), defined as follows:

$$\text{CONG}(f, \lambda, D) = \max_m \frac{\sum_{ij} f_{ij}(m) d_{ij}}{cap(m) \sum_{m \in I_i} \lambda_i}, \quad (6)$$

where $\sum_{ij} f_{ij}(m) d_{ij}$ is the aggregated load on link m for all s-d pairs, $\sum_{m \in I_i} \lambda_i$ is the aggregated portion of time available for link m to transmit from all independent sets it belongs to, $(\sum_{ij} f_{ij}(m) d_{ij}) / (cap(m) \sum_{m \in I_i} \lambda_i)$ is ratio between the aggregated load and the available capacity on link m, i.e., the congestion level of link m. The congestion of the network is then defined as the maximum congestion among all links. An optimal routing and scheduling with respect to TM D minimizes the maximum congestion level:

$$\text{OPT}(D) = \min_{(f,\lambda)} \text{CONG}(f, \lambda, D).$$

In other words, OPT(D) is the minimum congestion level possible for the given D.

The *competitive ratio* of an (f,λ) on a given TM D measures the performance of the given (f,λ) with respect to that of the optimal one on TM D. It is defined as:

$$\text{PERF}(f, \lambda, D) = \frac{\text{CONG}(f, \lambda, D)}{\text{OPT}(D)}.$$

The competitive ratio is always at least 1. It is exactly 1 if and only if the routing and scheduling is optimal for D.

Let $\mathbf{D}$ includes all possible TMs. The *oblivious ratio* is defined as

$$\text{PERF}(f, \lambda, \mathbf{D}) = \max_{D \in \mathbf{D}} \text{PERF}(f, \lambda, D).$$

In other words, the oblivious ratio is the worst performance ratio of the given (f, λ) with respect to all TMs. A routing and scheduling with the minimum oblivious ratio is the optimal oblivious-oblivious routing-and-scheduling. It optimizes the worst-case performance.

A. Routing and Scheduling with No Traffic Knowledge

Our objective is to find a traffic-oblivious routing-and-scheduling to minimize the oblivious ratio. In other words, we want a fixed routing and scheduling pair (f, λ) such that it achieves the worst-case minimum congestion level for all possible TMs under consideration, among all fixed routing-scheduling pairs. Formally, the optimal (f, λ) is defined as

$$\underset{(f,\lambda)}{argmin}\ \text{PERF}(f, \lambda, \mathbf{D}),$$

where $\mathbf{D}$ includes all the possible TMs under consideration. We first consider the case with no traffic information.

The optimal oblivious ratio r^* and the corresponding (f^*, λ^*) can be computed by solving the following problem:

$$\min_{(f,\lambda)} r \qquad (7)$$

$$\sum_{i=1}^{K} \lambda_i \leq 1$$

f is a routing

$\forall$ links m, $\forall$ TMs D with $\text{OPT}(D) = 1$:

$$\frac{\sum_{ij} f_{ij}(m) d_{ij}}{cap(m) \sum_{m \in I_i} \lambda_i} \leq r.$$

In this problem, if we scale a TM D by a constant, the performance ratio remains unchanged. So we only need to consider the D's with $\text{OPT}(D) = 1$ when we compute the oblivious ratio, as other D's are scaled versions of such matrices. The problem is interpreted as follows: consider any given TM D, if it takes its corresponding optimal routing and scheduling scheme one unit of time to serve the demand, then it takes (f^*, λ^*) *at most* r^* unit of time to serve the same demand.

We note that the last constraint is not a linear constraint. To solve the problem, we introduce an alternative variable $\beta_i = r\lambda_i$. We transform Eq. (7) to the following LP problem.

$$\min_{(f,\beta)} r \qquad (8)$$

$$\sum_{i=1}^{K} \beta_i \leq r$$

f is a routing

$\forall$ links m, $\forall$ TMs D with $\text{OPT}(D) = 1$:

$$\sum_{ij} f_{ij}(m) d_{ij} / cap(m) \leq \sum_{m \in I_i} \beta_i. \qquad (9)$$

We call this problem the master LP. After obtaining the optimal solution, we can obtain λ^* by $\lambda_i^* = \beta_i^* / r^*$. With some abuse of notation, we also call $\beta = \{\beta_i\}$ scheduling.

The master LP has infinite number of constraints because there are infinite number of D's with $\text{OPT}(D) = 1$. Next, we transform the LP formulation with infinite number of constraints to a LP formulation with finite number of constraints. Note that for a given link m, routing and scheduling pair (f,β), the constraint in Eq. (9) is satisfied if the maximum value of the following slave LP is less than 0.

$$\max_{g,\mu,d} \left(\sum_{ij} f_{ij}(m) d_{ij} / cap(m) - \sum_{m \in I_i} \beta_i \right) \qquad (10)$$

g is a flow of demand d_{ij} (11)

$$\sum_{i=1}^{K} \mu_i \leq 1 \qquad (12)$$

$$\forall \text{ links } l : \sum_{ij} g_{ij}(l) \leq \sum_{l \in I_i} \mu_i cap(l) \qquad (13)$$

$$\forall \text{ demands } i \to j\text{: } d_{ij} \geq 0, \qquad (14)$$

where both $g_{ij}(e)$ and μ_i are new variables introduced in the slave LP to restrict D. Note that g is the flow, defined as follows:

$$\forall \text{ edges } e, \forall \text{ nodes } i, j \neq i : g_{ij}(e) \geq 0 \qquad (15)$$

$$\forall \text{ nodes } i, j \neq i, k \neq i, j :$$

$$\sum_{e \in \text{out}(k)} g_{ij}(e) - \sum_{e \in \text{in}(k)} g_{ij}(e) = 0$$

$$\forall \text{ nodes } i, j \neq i :$$

$$\sum_{e \in \text{out}(i)} g_{ij}(e) - \sum_{e \in \text{in}(i)} g_{ij}(e) = d_{ij},$$

and μ_i is the portion of time the maximal independent set i can be active. With the two intermediate variables g and μ, only D's with $\text{OPT}(D) \leq 1$ satisfy the constraints defined in Eqs. (11-14). As stated in [12], at least one of the maximum of a linear objective function over a polyhedron are obtained on a vertex of the polyhedron and that polynomial time LP algorithms can obtain such a vertex maximum. Thus, the maximum objective value must be obtained at a certain D with $\text{OPT}(D) = 1$. Therefore, the slave LP can be used to check the constraint Eq. (9) in the master LP problem.

To further reduce the complexity, we combine the master LP with the slave LP. First, for the given m and

(f, β) pair, we derive the dual form of the slave LP (10):

$$\min_{s,\pi,p(m)} \left(p(m) - \sum_{m \in I_i} \beta_i \right) \quad (16)$$

$$\forall \text{ nodes } i, j \neq i : \frac{f_{ij}(m)}{cap(m)} \leq s_{ij}(m, j)$$

$$\forall \text{ nodes } i, j \neq i, \forall \text{ edges } e = i' \to j' :$$

$$\pi(m, \text{link-of}(e)) + s_{ij}(m, i') - s_{ij}(m, j') \geq 0$$

$$\forall \text{ maximal independent sets } k \in [1, K] :$$

$$\sum_{l \in I_k} \pi(m, l) cap(l) \leq p(m)$$

$$\forall \text{ links } l : \pi(m, l) \geq 0$$

$$\forall \text{ nodes } i, j \neq i, \forall \text{ nodes } k :$$

$$s_{ij}(m, k) \geq 0$$

$$\forall \text{ nodes } i, j \neq i : s_{ij}(m, i) = 0$$

$$p(m) \geq 0,$$

where link-of(e) denotes the link that edge e belongs to. Since we assume bidirectional link in our study, $l_{ij} =$ link-of(e_{ij}) = link-of(e_{ji}). The dual variable $s_{ij}(m, j)$ corresponds to the flow conservation constraint of d_{ij}, $\pi(m, l)$ to constraint (13), and $p(m)$ to constraint (12).

Based on the duality theorem [18], the dual LP has the same objective value as the primal LP problem. We then combine the master LP (8) with the dual form of the slave LP (16) to form a single LP:

$$\min r \quad (17)$$

$$f \text{ is a routing}$$

$$\sum_{i=1}^{K} \beta_i \leq r$$

$$\forall \text{ links } m : p(m) - \sum_{m \in I_i} \beta_i \leq 0$$

$$\forall \text{ links } m, \forall \text{ nodes } i, j \neq i :$$

$$\frac{f_{ij}(m)}{cap(m)} \leq q_m(i, j)$$

$$\forall \text{ links } m, \forall \text{ nodes } i, \forall \text{ edges } e = j \to k :$$

$$\pi(m, \text{link-of}(e)) + q_m(i, j) - q_m(i, k) \geq 0$$

$$\forall \text{ links } m, \forall \text{ maximal independent sets } k \in [1, K] :$$

$$\sum_{l \in I_k} \pi(m, l) \leq p(m)$$

$$\forall \text{ links } m, l : \pi(m, l) \geq 0$$

$$\forall \text{ links } m : p(m) \geq 0$$

$$\forall \text{ links } m, \forall \text{ nodes } i : q_m(i, i) = 0$$

$$\forall \text{ links } m, \forall \text{ nodes } i, j \neq i : q_m(i, j) \geq 0$$

$$\forall \text{ maximal independent sets } k \in [1, K] : \beta_k \geq 0.$$

In this LP, there are finite number of variables and finite number of constraints. Thus, it can be solved using standard LP solvers.

While motivated by previous work, our problem is different because we need to explicitly address schedulability. In [11], [12], [13], Internet routing is considered, and thus there is no interference or schedulability issue. We note that we cannot directly apply the schedulability constraint defined in Eq. (2) to the slave LP in [13] to set additional constraints on $g_{ij}(e)$. Because schedulability constraint is not considered in the master LP of [13], some qualified D's (i.e., with OPT$(D) = 1$) may not be included if (2) is directly applied. As a consequence, the maximum value of the slave LP could be obtained at some D's with OPT$(D) \neq 1$, which would result in incorrect solutions.

In [14], the authors propose an energy-efficient traffic-oblivious routing scheme in wireless sensor networks. The authors also discuss the schedulability issue and suggest additional constraints to be placed in a slave LP formulation. In [14], the definition of OPT(D) is based on power consumption instead of link congestion, and thus does not need to take into account schedulability. Therefore, the schedulability constraint can be directly applied to the slave LP as suggested in the discussion in [14]. However, applying schedulability constraint in the slave LP implies that schedulability is guaranteed only with perfect traffic information, which deviates from the original motivation of being traffic-oblivious. We note that schedulability is not a major concern in wireless sensor networks due to the low traffic demand. In comparison, because our objective is to maximize network utilization, schedulability constraint has to be defined in OPT(D). Therefore, the approach in [14] does not apply to our case. In addition, our scheduling is also traffic-oblivious while it is not in [14]. In summary, all previous works consider traffic-oblivious *routing* and our framework considers traffic-oblivious *routing-and-scheduling*, which addresses the unique constraint in wireless networks.

B. Routing and Scheduling with Some Traffic Information

The LP in (17) is derived without any traffic information. As discussed in Section II-B, in some cases we could also have a certain amount of traffic information, i.e., the active s-d pairs or the s-d pairs with the range of the traffic demand. Such information can be modelled by introducing constraints on d_{ij}; i.e., $a_{ij} \leq d_{ij} \leq b_{ij}$ for s-d pairs, and $d_{ij} = 0$ for non-s-d pairs. Similar to [13], we introduce slack variables $s_m^-(i,j)$ and $s_m^+(i,j)$ for the lower and upper bounds on d_{ij}. Following similar approaches in Section III-A, we can find the optimal routing-scheduling by solving the following finite LP formulation.

$$
\begin{aligned}
&\min_{f,\beta} r \qquad (18)\\
&f \text{ is a routing}\\
&\sum_{i=1}^{K} \beta_i \le r\\
&\forall \text{ link } m: p(m) - \sum_{m \in I_i} \beta_i \le 0\\
&\forall \text{ link } m, \forall \text{ node } i, j \ne i:\\
&\quad \frac{f_{ij}(m)}{cap(m)} - s_m^+(i,j) + s_m^-(i,j) = q_m(i,j)\\
&\forall \text{ link } m, \forall \text{ node } i, \forall \text{ edge } e = j \to k:\\
&\quad \pi(m, \text{link-of}(e)) + q_m(i,j) - q_m(i,k) \ge 0\\
&\forall \text{ link } m, \forall \text{ maximal independent sets } k \in [1,K]:\\
&\quad \sum_{l \in I_k} \pi(m,l) \le p(m)\\
&\forall \text{ link } m: \sum_{ij} \left(b_{ij} s_m^+(i,j) - a_{ij} s_m^-(i,j)\right) \le 0\\
&\forall \text{ links } m, l: \pi(m,l) \ge 0\\
&\forall \text{ link } m: p(m) \ge 0\\
&\forall \text{ link } m, \forall \text{ node } i: q_m(i,i) = 0\\
&\forall \text{ link } m, \forall \text{ node } i, j: q_m(i,j) \ge 0\\
&\forall \text{ maximal independent sets } k \in [1,K]: \beta_k \ge 0\\
&\forall \text{ link } m, \forall \text{ node } i, j: s_m^-(i,j) \ge 0\\
&\forall \text{ link } m, \forall \text{ node } i, j: s_m^+(i,j) \ge 0.
\end{aligned}
$$

Note that, if $a_{ij} = b_{ij} = d_{ij}$ for all i, j (i.e., there is no uncertainty on traffic information), the oblivious ratio is one, which means that the routing and scheduling outputted from our framework is the optimal routing and scheduling.

IV. Performance Evaluation

A. Simulation Setup

We use Qualnet [19] for performance evaluation. Qualnet is a commercial wireless-network simulator that provides good PHY layer modeling as well as upper-layer protocol implementations. We use 802.11a/g PHY layer. The data rate is 54 Mbps. Every node uses the same transmission power. The channel model is the 2-ray model with slow fading. The conflict graph is measured after the network is setup. For each link A, we enable all the other links one by one, and measure the throughput. If the throughput degrades after link B is enabled, A and B interfere with each other. CBR is used as the application for each active s-d pair. We use the maximum possible payload size (i.e., 1500 bytes MAC layer payload). The traffic demand varies, and will be explained later in this section. For a multipath routing, a node splits its traffic among multiple paths based on the corresponding ratio on each path. We implement the scheduling using TDMA with fixed length super frames where each time slot is allocated to a maximal independent set. Total number of time slots allocated to a maximal independent set i is proportional to its time allocation λ_i. Note that the length of the super frame is carefully chosen to fit the precision of the scheduling λ_i. For example, if the λ_i is rounded to the second decimal place, we will use 100 time slots in each super frame. Within a super frame, we schedule the transmission as fair as possible. For instance, consider two maximal independent sets A and B, A is allocated 60 time slots while B is allocated 40 time slots. In the first 80 time slots, we schedule A and B in a round-robin manner, then A transmits in the remaining 20 time slots.

To make the scheduling overhead comparable to the 802.11 CSMA-based MAC protocol, the slot duration is carefully chosen such that for the simplest scenario where one node is always sending packets to the other node with maximum possible payload size, both 802.11 and scheduling lead to the same throughput. In other words, the slot duration includes all the MAC and physical overhead of 802.11 except for the collision overhead. This enables us to distinguish the scheduling gain (i.e., avoiding collision) from joint routing-and-scheduling gain.

We have conducted simulation in a variety of topologies, including 9-node grid, 16-node grid, chain, and random networks. We have also conducted simulation under different traffic patterns, including single-sink, double-sink, random peer-to-peer, and all s-d pairs. Due to space limitation, we present the simulations in a 16-node grid with single sink traffic pattern unless otherwise specified. Other simulations yield the same conclusions. In the single sink traffic pattern, there is one sink node in the network and every other node communicates with the sink node. This traffic pattern is common in wireless mesh networks where there is a mesh portal providing Internet access to the other nodes in the network. We compare the performance of four routing and scheduling schemes:

- *AODV+802.11*: The routing protocol is AODV, and the MAC is 802.11 CSMA-based MAC.

- *AODV+schd*: AODV is combined with an optimal scheduling for a given TM D. The optimal scheduling is calculated as follows:

$$\min \sum_{i=1}^{K} \beta_i \tag{19}$$

$\forall$ links m:

$$\sum_{ij} f_{ij}(m) d_{ij}/cap(m) \leq \sum_{i:m\in I_i} \beta_i,$$

where $f_{ij}(m)$ is determined by AODV routing.

- *estimate*: We assume a *base* TM, which is our best estimate of the actual traffic. The actual traffic pattern and demand may vary from the estimation. Then we compute the optimal routing and scheduling based on this estimated traffic matrix.
- TORS: The oblivious routing and scheduling we proposed in Section III. We assume the same base TM as in *estimate*, but takes into account traffic information uncertainty.

We assume a base traffic demand for each s-d pair. The actual traffic demand fluctuates around the base traffic demand. In order to model the traffic fluctuation as well as traffic information inaccuracy, we consider the error margin w. An actual TM, $D' \in$ **D'**, satisfies $d_{ij}/w \leq d'_{ij} \leq w d_{ij}$. For example, if $w = 1.5$, the upper bound on traffic demand is 225% of the lower bound. For each set of simulation, the actual demand is generated randomly within $[d_{ij}/w, d_{ij}{*}w]$. We do not use the more intuitive range $[d_{ij}-w, d_{ij}+w]$. This is because we want the lower bound to be always larger than zero. As we explained in Section III-A, the oblivious ratio remains the same if we scale TM D by a constant. Thus, once the lower bound is zero, no matter what the upper bound is, it is always equivalent to no information at all. In each setting, we run 100 realizations, each lasts for 60 seconds. In each realization, we pick a random traffic demand within the range for each s-d pair. This simulates the traffic variations during a long-term connection, e.g., 100 minutes and the traffic demand changes every minute. We measure the average and worst case throughput performance over the 100 realizations.

B. Traffic-Oblivious Routing and Scheduling

We compare the performance of the four routing and scheduling schemes discussed in Section IV-A. The results are tabulated in Table I. Each row corresponds to a w, i.e., the level of traffic variations. In the second and third rows, the left entry is the average throughput, and the right one is the minimum throughput (i.e., worst case performance), both obtained from a set of 100 simulations.

TABLE I

COMPARE FOUR ROUTING AND SCHEDULING SCHEMES

w	*AODV+802.11*	*AODV+schd*	*Estimate*	TORS
1.0	8.85	11.58	15.15	15.15
1.5	8.04 / 7.35	8.62 / 7.88	11.50 / 10.04	11.73 / 10.35
2.5	7.73 / 7.23	8.42 / 6.35	10.65 / 8.23	10.46 / 8.58

Both *estimate* and TORS achieve higher throughput than *AODV+802.11*, as they both benefit from the spatial reuse and scheduling. *AODV+schd* performs in between, which benefits from scheduling. In the first row, since $w = 1.0$, there is no variation in the traffic demand, which means that perfect traffic information is given. In this case, TORS and *estimate* both achieve optimal performance. When there is uncertainty in the traffic demand, e.g., $w = 1.5$ as in the second row, TORS achieves a higher worst-case performance. The trend is more obvious when the uncertainty increases, i.e., $w = 2.5$ as in the third row. TORS leads to a slightly lower average throughput, but a considerably higher worst-case throughput. This is in accord with the objective of TORS — to achieve optimal worst-case performance. It also indicates that TORS achieves a smoother throughput when traffic demand varies.

C. Oblivious Routing with 802.11 MAC

Centralized routing is easier to implement than centralized scheduling because scheduling requires synchronization. Therefore, a natural step is to integrate centralized routing with distributed scheduling or sensing-based medium access scheme. Our heuristic is to generate the joint routing and scheduling using TORS and then combine routing with a decentralized MAC (e.g., 802.11). In our simulation, we observe that the performance of the heuristic depends on traffic patterns. If the number of s-d pairs is small, the proposed heuristics outperforms *ADOV+802.11* significantly. It is due to the spatial reuse introduce by the routing. On the other hand, if there is a large number of s-d pairs with heavy load, such as in the all-pair traffic pattern, the heuristic using TORS performs worse than *AODV+802.11* due to the existence of long routes and the lack of optimal scheduling support.

TABLE II
SINGLE SINK TRAFFIC PATTERN: OBLIVIOUS ROUTING WITH 802.11 MAC

w	*AODV+802.11*	Oblivious routing+802.11
1.0	8.85	9.81
1.5	8.00 / 7.12	9.58 / 9.35
2.5	7.69 / 7.00	9.35 / 8.77

TABLE III
PERFORMANCE UNDER LINK FLUCTUATIONS

Packet loss rate	*AODV+802.11*	TORS
0.2%	8.31	11.46
3.2%	7.46	10.31
4.4%	7.15	9.88
5.7%	6.96	9.50
7.3%	6.65	8.96

TABLE IV
SINGLE SINK TRAFFIC PATTERN: THE OBLIVIOUS RATIO AND THROUGHPUT UNDER DIFFERENT W'S

w	1.0	1.5	2.0	2.5	3.0	4.0
obl. ratio	1.00	1.24	1.52	1.64	1.71	1.79
TORS	0.696	0.630	0.596	0.462	0.458	0.456
ideal	0.696	0.689	0.697	0.697	0.693	0.698

TABLE V
PERFORMANCE UNDER INACCURATE ESTIMATION OF THE TRAFFIC

	Single-sink	Random	All-pair
estimate	0.18 / 0.16	0.39 / 0.31	0.76 / 0.62
TORS	0.27 / 0.24	0.51 / 0.43	0.74 / 0.70

D. Performance under Link Fluctuations

We also verify the robustness of TORS under link fluctuations. We introduce the link fluctuations by adjusting the shadowing effect in the physical layer. We measure the throughput of AODV+802.11 and TORS, as well as the physical layer packet loss rate. The results are tabulated in Table III. The performance degradation of TORS is comparable to *AODV+802.11* as packet loss rate increases.

Although not shown here, we also observe that, in an environment with very severe fading, TORS performs much more robust than *AODV+802.11*. The reason is that there are always some of the links which are considered broken by AODV in such an environment. Thus it has to frequently re-route its traffic by performing route discovery procedures. On the other hand, oblivious routing does not have route discovery overhead. Furthermore, oblivious routing is of multiple paths. Its performance is thus robust to random broken links.

E. Insights of TORS

In order to better understand TORS, we measure the oblivious ratio obtained under different levels of traffic variations. In addition, we compare the performance between oblivious and an ideal but not realistic routing and scheduling scheme. For the *ideal* strategy, we assume that the perfect traffic information is always available. Thus one can obtain the optimal routing-scheduling, and thus the optimal throughput for each traffic instance. Its performance serves as the benchmark.

The results are shown in Table IV. The first row is w. The second row is the corresponding oblivious ratio for TORS. The third and fourth rows are the average throughput for TORS and *ideal* schemes, respectively. As w increases, traffic information uncertainty increases. The performance of *ideal* is stable because it assumes perfect traffic information. In comparison, TORS has to cope with more variation in traffic demands, which leads to a higher oblivious ratio, and lower average throughput as w increases. The results in Table IV indicates that TORS achieves satisfactory worst case performance (indicated by the oblivious ratio) and average performance, compared to the *ideal* case.

TORS is also robust under false traffic information. In Table V, we compare TORS and *estimate*. Both calculate routing and scheduling assuming the all-pair traffic pattern [2] with $w = 1.5$. However, in the simulations, the actual traffic patterns are single-sink, random [3], and all-pair, respectively. For each entry in the table, the left is the average throughput, and the right is the worst-case throughput.

We observe that, when the estimation is correct, i.e., the all-pair traffic pattern, TORS achieves comparable average performance and better worst-case performance. In addition, when the actual traffic patterns are single-

[2] All-pair traffic pattern: every node transmits to all the other nodes.
[3] Random pattern: we randomly pick half of the nodes as senders. Each node randomly chooses a destination.

sink and random (i.e., largely deviate from the estimation), the performance of TORS is robust and always better than *estimate*. It indicates the robustness of TORS even under false traffic information.

We study in detail the output of TORS to understand the desirable properties of robust routing and scheduling. The routing generated by TORS is multi-path routing that *exploits spatial reuse* in the network and *tolerance to network dynamics*. In general, TORS results in a more even distribution of flows in the network and more balanced utilization of network resources. Its tolerance to traffic dynamics is due to the load balancing effect of splitting flows among multiple paths. In addition, TORS tends to allocate overlapping routes to different flows; i.e., an edge can be used by subflows from multiple flows. Such an allocation provides tolerance to traffic dynamics due to the smoothing effect of statistic multiplexing. In addition, TORS makes it easier to admit new flows because it adapts to demand changes and distributes resource relatively evenly.

V. Conclusion

In this paper, we propose TORS, a joint routing-and-scheduling scheme, that achieves robust performance under traffic information uncertainty. To elaborate, the proposed scheme optimizes the worst-case performance (in terms of relative congestion) for a range of traffic conditions. The proposed scheme works under the whole spectrum of traffic information uncertainty from perfect traffic information to no traffic information. The performance TORS adapts to the granularity of the traffic information available. The more accurate the information, the better the performance. We show that TORS achieves optimal worst-case performance, as well as good average performance, even compared to the *ideal* case. The performance of TORS is robust under traffic dynamics, or even false traffic information.

TORS generates multipath routes that exploit spatial reuse and provide more balanced utilization of network resource. Because of these properties, TORS achieves robust performance against traffic dynamics and link quality fluctuations. These properties also provide insights into the design joint routing and MAC schemes in a distributed manner.

Acknowledgment

The work was in part supported by NSF through CAREER Award #0448613 and Grant #0520126, and by Intel through a gift grant

References

[1] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *ACM MobiCom*, 2003.

[2] R. Cruz and A. Santhanam, "Optimal routing, link scheduling and power control in multihop wireless networks," in *IEEE INFOCOM*, 2003.

[3] J. Zhang, H. Wu, Q. Zhang, and B. Li, "Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks," in *IEEE Broadband Networks*, 2005.

[4] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2005, pp. 73–87.

[5] T. Larsson and N. Hedman, "Routing protocols in wireless ad-hoc networks - a simulation study," Master's Thesis, LuLea Tekniska University, 1998.

[6] J. Bicket, D. Aguayo, S. Biswas, and R. Morris, "Architecture and evaluation of an unplanned 802.11b mesh network," in *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2005, pp. 31–42.

[7] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM Press, 2004, pp. 133–144.

[8] S. Mueller, R. P. Tsang, and D. Ghosal, "Multipath routing in mobile ad hoc networks: issues and challenges," in *Invited paper in Lecture Note in Computer Science*, 2004.

[9] W. Lou, W. Liu, and Y. Zhang, "Performance optimization using multipath routing in mobile ad hoc and wireless sensor networks," in *Combinatorial optimization in communication networks*, 2005.

[10] Y. Ganjali and A. Keshavarzian, "Load balancing in ad hoc networks: single-path routing vs. multi-path routing," in *IEEE Infocom*, 2004.

[11] H. Räcke, "Minimizing congestion in general networks," in *IEEE FOCS*, 2002.

[12] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Racke, "Optimal oblivious routing in polynomial time," in *ACM STOC*, 2003.

[13] D. Applegate and E. Cohen, "Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs," in *ACM SIGCOMM*, 2003.

[14] Y. Li, J. Harms, and R. Holte, "Optimal traffic-oblivious energy-aware routing for multihop wireless networks," in *IEEE INFOCOM*, 2006.

[15] Cisco, "Using radio resource management to deliver secure and reliable wlan services," White Paper.

[16] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and qos routing in multi-channel wireless mesh networks," in *ACM MobiHoc*, 2005.

[17] M. Burkhart, P. von Rickenbach, R. Wattenhofer, and A. Zollinger, "Does topology control reduce interference," in *ACM MobiHoc*, 2004.

[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Second Edition*. The MIT Press, 2001.

[19] I. Scalable Network Technologies, "Qualnet 3.9 user's guide," 2005.