Providing Guaranteed Protection in Multi-Hop Wireless Networks with Interference Constraints

Greg Kuperman and Eytan Modiano, *Fellow, IEEE*

Abstract—We consider the problem of providing protection against failures in wireless networks subject to interference constraints. Typically, protection in wired networks is provided through the provisioning of dedicated backup paths. This approach has not been previously considered in the wireless setting due to the prohibitive cost of backup capacity. Assigning capacity for dedicated backup paths in a wireless setting can more than double the total resources required from what was needed without protection, which can make protection infeasible. However, we show that in the presence of interference, guaranteed protection can be provided for all demands with little, and oftentimes *no*, additional resources beyond what was required without any protection. This is due to the fact that after a failure, links that previously interfered with the failed link can be activated, thus leading to a "recapturing" of lost capacity. We provide an ILP formulation to find an optimal solution for both binary and SINR interference constraints, and develop corresponding time-efficient algorithms. Our approach utilizes up to 87 percent less protection resources than traditional disjoint path routing to provide guaranteed protection. For the case of 2-hop interference, our protection scheme requires only 8 percent more resources on average than providing no protection whatsoever.

Index Terms—Network-level security and protection, network architecture and design, network communications

1 Introduction

Multi-hop wireless mesh networks have become increasingly ubiquitous, with wide-ranging applications from military to sensor networks. As these networks continue gaining in prominence, there is an increasing need to provide protection against node and link failures. Wireless mesh networks have recently emerged as a promising solution for providing Internet access, particularly in developing nations [1]. Since these networks will be tightly coupled with the wired Internet to provide Internet services to end-users, they must be equally reliable. Failures in wireless networks can occur due to node failure, obstructions, deep fades, as well as malicious attacks [2]. Wired networks have long provided pre-planned backup paths, which offer rapid and guaranteed recovery from failures. These protection techniques cannot be directly applied to wireless networks due to interference constraints. As opposed to wired networks, two wireless nodes in close proximity will interfere with one another if they transmit simultaneously in the same frequency channel. In addition to finding a backup route to provide protection against failure, an interference-free schedule of link transmissions needs to be specified. In this work, we consider the problem of providing guaranteed protection in wireless networks with interference constraints via pre-planned backup routes, as well as their corresponding link transmission schedules.

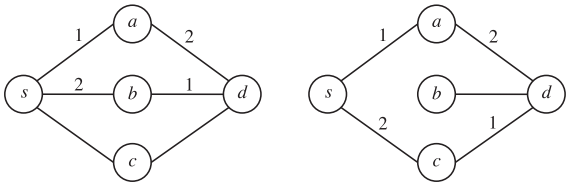
• *The authors are with LIDS, Massachusetts Institute of Technology, Cambridge, MA 02139. E-mail: {gregk, modiano}@mit.edu.*

Manuscript received 30 Apr. 2017; revised 10 Dec. 2016; accepted 17 Apr. 2017. Date of publication 19 Apr. 2017; date of current version 1 Nov. 2017. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMC.2017.2696001

Guaranteed protection schemes for wired networks have been studied extensively [3], [4], [5], [6], [7], [8], with the most common scheme being 1:1 disjoint path protection [7]. The 1:1 protection scheme provides an edge or node disjoint backup path for each primary path, and guarantees the full demand to be available at all times after any single link failure. Since the protection resources are preallocated, recovery from a failure is rapid and guaranteed. Disjoint path protection is also resource efficient: Failure disjoint primary paths will not fail simultaneously for any single failure, and hence can share the same set of backup resources. This backup sharing can significantly reduce the protection resources needed [4], [9]. In addition, disjoint path protection is non-disruptive in the sense that only the demands that fail will be switched to their backup paths, allowing the connections that did not fail to continue using their primary paths.

Protection schemes optimized for wireless networks with interference constraints have not yet been considered. Typically, an approach for resiliency in wireless networks (in particular sensor networks) is to ensure that there exists "coverage" for all nodes given some set of link failures [10], [11]. This approach to resiliency does not consider routing and scheduling with respect to interference constraints, and assumes that there exists some mechanism to find a route and schedule at any given point in time. Furthermore, there is no guarantee that sufficient capacity will be available to protect against a failure. The idea of applying 1:1 disjoint path protection in wireless networks is briefly mentioned in [12]. However, [12] does not study the specific technical details of such an approach to wireless protection.

The goal of this paper is to study protection mechanisms for wireless networks with a particular focus on the impact



(a) Before a failure (b) After $\{s, b\}$ fails

Fig. 1. Time slot assignment for protection in a wireless network.

of wireless interference and the need for scheduling. The addition of interference constraints makes the protection problem in a wireless setting fundamentally different from the ones found in a wired context. After a failure in a wireless network, links that could not have been used due to interference with the failed link become available, and can be used to recover from the failure. In fact, it is often possible to add protection in a wireless setting without using any additional resources. This paper is an extended version of our preliminary work on this topic [13], [14].

Consider allocating a protection route for the example shown in Fig. 1. Any two nodes within transmission range have a link between them. The wireless network operates in a time-slotted fashion, with equal length time slots available for transmission; each link's time slot assignment is shown in the figure. For this example, we assume a 1-hop interference model where any two links that have a node in common cannot be active at the same time. Additionally, we assume unit capacity links. Before any failure, the maximum flow from s to d is 1, and can be achieved using a schedule of two time slots, as shown in Fig. 1a. At any given point in time, only one outgoing link from s can be active, and similarly, only one incoming link to d can be active. Wireless links $\{s, c\}$, and $\{c, d\}$ cannot be used prior to the failure of $\{s, b\}$, but these links become available after $\{s, b\}$ fails. After the failure of $\{s, b\}$, flow can be routed from s to c during time slot 2, and from c to d during slot 1, as shown in Fig. 1b. Similar schedules can be found for failures of the other links. The maximum flow from s to d is 1 for both before *and* after a failure; i.e., there is no reduction in maximum throughput when allocating resources for a guaranteed protection route on $\{s, c\}$ and $\{c, d\}$: Protection can be assigned for "free". This is in contrast to a wired network where the maximum throughput without protection from s to d is 3, and the maximum throughput when assigning a protection route on $\{s, c\}$ and $\{c, d\}$ is 2, which amounts to a $\frac{1}{3}$ loss in throughput due to protection.

The novel contributions of this paper is introducing the Wireless Guaranteed Protection (WGP) problem in multi-hop networks with interference constraints. Our goal is to provide protection in wireless networks that is similar to the popular disjoint path approach in wired networks. In particular, we examine the problem of providing protection by using disjoint path protection for wireless networks that are subject to interference constraints that is both non-disruptive and resource-efficient. The main insight we take advantage of is that interference in wireless networks can be used for greater resource efficiency: resources that are freed after a failure in the network can be reused for protection from that failure. We formulated an ILP to solve WGP, giving solutions that used 87 percent fewer protection resources on average than the wired disjoint path scheme in wireless networks. We then developed a time-efficient algorithm for WGP that performed almost as well as the ILP solution and has significantly faster run-time. We consider both binary and SINR interference constraints, allowing our wireless protection scheme to be used for almost any standard interference model.

The outline of the paper is as follows. In Section 2, the model for WGP is presented. In Section 3, an optimal solution is developed via an integer linear program for general interference constraints. In Section 4, time-efficient algorithms are developed that perform within 4.5 percent of the optimal solution.

2 Model and Problem Description

In this paper, solutions to the guaranteed protection problem for multi-hop wireless networks subject to interference constraints are developed and analyzed. Our goal is to provide protection in a manner similar to what has been done in the wired setting. Namely, after the failure of some network element, all connections must maintain the same level of flow that they had before the failure. In order to do so, resources are allocated and scheduled in advance on alternate (backup) routes to protect against failures.

Wireless networks are inherently different than wired networks. In a wired network, two adjacent nodes can transmit simultaneously because they do not interfere with one another; if capacity exists on a set of links, a path can be routed using that capacity. Wireless networks are different; interference constraints must be considered. A set of links in close proximity cannot transmit simultaneously on the same frequency channel; only one link from that set can be active at a time, or else they will interfere with one another. Not only must a path between the source and destination be found with available capacity, but also a schedule of link transmissions needs to be determined. This is known as the routing and scheduling problem [12], [15], [16], [17], [18], [19], [20], [21], [22], which is known to be NP-Hard [15].

The addition of interference constraints adds complexity to the traditional wired protection problem, but also presents an opportunity to gain protection from failures with minimal loss of throughput. After a failure in a wireless network, links that could not have been used due to interference with the failed link become available, and can be used to recover from the failure. In fact, it is often possible to add protection in a wireless setting without the need for any additional resources that result in a loss of throughput; i.e., protection can be provided *for free*.

Our goal is to provide disjoint backup path protection in a manner similar to what has been done in the wired setting. Namely, after the failure of some network element, connections that fail switch to their respective backup paths, and connections that did not fail continue to use their primary paths. After any failure, connections must maintain the same amount of flow that they had before the failure. As opposed to wired networks, transmissions in wireless networks must be scheduled so communications can occur without interference. For connections that were not affected by a failure, in addition to continuing to use their primary paths, they must also maintain the same transmission schedule. These requirements guarantee

minimal disruption to the connections unaffected by the failure, and thus provide rapid recovery. In order to meet these requirements, resources are allocated and scheduled in advance for both the primary and backup routes to protect against failures.

The mechanism for disjoint path protection in wireless networks is as follows. Each demand will have a primary and backup path, as well as an interference-free schedule for those paths, and after the failure of some network element, a demand whose primary path fails will switch to its disjoint backup path and schedule. If a demand's primary path did not fail, it will continue to use its pre-failure primary path and schedule. If a primary path does fail, the time slots that were used to schedule that path are no longer needed, and can be reused for the protection path.

We consider both binary and SINR interference constraints in their general form. In the binary interference model, for any pair of links, $\{i,j\}$ and $\{k,l\}$, either both links can be active simultaneously, or at most one link can be active [23]. Binary interference is used for the K-hop interference model [24], and the protocol interference model [25]. In K-hop interference, if link $\{k,l\}$ is within K hops of link $\{i,j\}$, the two links will interfere. In the protocol model, link $\{i,j\}$ can be active only if i is within range of j, and no other nodes that are within range of j are transmitting. Other examples of binary interference exist, notably [16] that uses a variation of the protocol model.

In the SINR interference model, interference is determined by examining the cumulative effect that all transmitting nodes in the network have on some receiver. If node i wishes to communicate to node j, then the power received at j from i must be sufficiently greater than all the other nodes transmitting at the same time, as well as the ambient noise [25]. We define the following: P_{ij} is the received power at node j when node i is transmitting; β is the reception threshold; N is the ambient noise; u is a transmitting node, v is a receiving node; V' is the set of nodes that are currently transmitting, excluding u. Using the SINR interference model, for a transmission from node u to node v to be successfully received, the following condition must be met: $P_{uv} \geq \beta(N + \sum_{v' \in V'} P_{v'v})$. To keep our SINR interference model general, we allow the received power from i to j, P_{ij}, to take any value. This model was used in [26], and can accomodate any power allocation scheme. The physical interference model is a common version of SINR [27]. Define r_i as node i's transmit power, $d(i,j)$ as the distance between nodes i and j, and α as the path-loss exponent, where α is commonly between 2 and 6. Using these values, the received power for the physical interference model is $P_{ij} = \frac{r_i}{d(i,j)^\alpha}$.

The following network model is used for the remainder of the paper. We are given a graph G with a set of wireless nodes V and edges E. A set of demands $(s_i, d_i) \in D$ must be routed and scheduled, such that there will exist a primary and disjoint backup path from s_i to d_i, $\forall i$. Since we are considering wireless networks in the context of backbone and last-mile services, we assume that the wireless nodes are static. For any node, we assume that its neighbors are fixed; hence, the set of edges E is fixed. For the binary interference model, an interference matrix $\mathcal{I}$ can be defined where $I_{ij}^{kl} \in \mathcal{I}$ is 1 if links $\{i,j\}$ and $\{k,l\}$ can be activated simultaneously (do not interfere with each other), and 0 otherwise. For the SINR interference model, a power matrix $\mathcal{P}$ can be defined where $P_{ij} \in \mathcal{P}$ is the received power at node j when node i is transmitting. Both $\mathcal{I}$ and $\mathcal{P}$ are fixed and known. We assume that the network uses a synchronous time slotted system, with equal length time slots, where the set of time slots used is $\mathcal{T}$, and $T = |\mathcal{T}|$.

Our objective is to minimize the total number of time slots needed to route and schedule all demands using disjoint path protection. Minimizing the length of the schedule will allow each link to communicate for a longer period of time, raising the overall throughput [15], [28]. Solutions are developed for both node and link failures, and similar to the work in wired protection, we use a single failure model, where we assume at most one failure at a time. Our work can be extended to multiple failures by considering additional disjoint paths. All transmissions share a single frequency channel. For now, we assume centralized control; the algorithms presented can be modified to work in a distributed fashion, as done in [29].

3 An Optimal Formulation for Wireless Guaranteed Protection

This section provides a mathematical formulation to the optimal solution for the Wireless Guaranteed Protection problem with general interference constraints. In particular, for a set of demands, a route and schedule needs to be found such that after any link failure, all end-to-end connections maintain their same level of flow. We first demonstrate that finding a minimum length schedule for WGP is NP-hard, and that the problem remains NP-hard even to approximate.

Theorem 1. *For Wireless Guaranteed protection with either binary of SINR interference constraints, determining a minimum-length schedule is NP-hard, and remains NP-hard to approximate.*

The proof can be found in Appendix A. Since WGP is NP-hard both to solve optimally and to approximate, an integer linear program (ILP) is formulated to find an optimal solution. In Section 3 an ILP is presented to find the optimal solution using both general binary and SINR interference constraints. In Section 3.2, a simulation of WGP is presented, and the results are compared to the use of a traditional wired 1:1 protection scheme in wireless networks.

3.1 Integer Linear Program for WGP

In wired networks, a typical objective function for protection is to minimize the total allocated capacity needed to satisfy all demands. A similar objective cannot be clearly defined for wireless networks since the concept of capacity changes in the presence of interference constraints. Consider some active link $\{i,j\}$. An adjacent link $\{j,k\}$ cannot be used simultaneously with $\{i,j\}$ because of interference; hence, simply adding additional link capacity (in a wired sense) will not enable its use. Another time slot must be allocated to allow a connection to use $\{j,k\}$ such that it does not interfere with $\{i,j\}$. Adding an additional time slot will reduce the time that each individual time slot in the schedule is active, which reduces the overall throughput of the

network [12], [15], [19]. For example, consider a network with two time slots and a connection that supports a flow of 1 using these two time slots. If a third time slot is added to the schedule, then the original two time slots are only active for $\frac{2}{3}$ of the total time, and that flow's scheduled throughput is reduced from 1 to $\frac{2}{3}$. Thus, the objective we consider is to use a minimum number of time slots to route and schedule each demand with protection. Further reading on efficient ILP formulations for wireless network design and protection can be found in [30], [31], [32].

The conditions for both link-disjoint and node-disjoint paths are given, where link-disjoint paths are guaranteed to only survive a link failure, and node-disjoint are guaranteed to survive either a link or node failure. We assume a set of demands D, where some demand between nodes s and d has its own throughput requirement f^{sd}.

For the ILP, the following values are given:

- $G = (V, E)$ is the graph with a set of vertices and edges
- D is the set of demands
- f^{sd} is the flow required between nodes (s, d); $f^{sd} \geq 0$
- u_{ij} is the capacity of link $\{i, j\}$
- $\mathcal{T}$ is the set of time slots in the system, $\mathcal{T} \subset \mathbb{Z}^+$
- M is a large constant
- Binary interference
  - $\mathcal{I}$ is the interference matrix, where $I_{ij}^{kl} \in \mathcal{I}$ is 1 if links $\{i, j\}$ and $\{k, l\}$ can be activated simultaneously, 0 otherwise
- SINR interference
  - $\mathcal{P}$ is the power matrix, where $P_{ij} \in \mathcal{P}$ is the received power at node j when node i is transmitting
  - β is the reception threshold
  - N is the ambient noise

The ILP solves for the following variables:

- x_{ij}^{sd} is 1 is the primary flow assigned for demand (s, d) on link $\{i, j\}$, 0 otherwise
- y_{ij}^{sd} is 1 is the protection flow assigned on link $\{i, j\}$ for demand (s, d), 0 otherwise
- $\lambda_{ij}^{sd,t}$ is a scheduling variable for the primary flow for demand (s, d) and is 1 if link $\{i, j\}$ is activated in time slot t, 0 otherwise
- $\delta_{ij,kl}^{sd,t}$ is a scheduling variable for the flow after the failure of link $\{k, l\}$ for demand (s, d), and is 1 if link $\{i, j\}$ is activated in time slot t, 0 otherwise
- τ_{ij}^{t} is a scheduling variable and is 1 if link $\{i, j\}$ is activated in time slot t, 0 otherwise
- $\pi_{ij,kl}^{t}$ is a scheduling variable, and is 1 if link $\{i, j\}$ is activated in time slot t after link $\{k, l\}$ fails, 0 otherwise
- s^t is 1 if time slot t is used by the primary or protection flow, and 0 otherwise

The objective function is to minimize the number of time slots (the length of the schedule) needed to route all demands with disjoint path protection

$$\textbf{Objective:} \quad \min \sum_{t \in \mathcal{T}} s^t. \tag{1}$$

The following constraints are imposed to find a feasible routing and scheduling.

Before a Failure:

- Find a primary path for demand (s, d) before any link failure

$$\sum_{\{i,j\} \in E} x_{ij}^{sd} - \sum_{\{j,i\} \in E} x_{ji}^{sd} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = d \\ 0 & \text{otherwise,} \end{cases} \quad \begin{matrix} \forall (s,d) \in D \\ \forall i \in V \end{matrix} \tag{2}$$

- Ensure a link is scheduled to support the primary flow for demand (s, d) on edge $\{i, j\}$

$$x_{ij}^{sd} \leq \sum_{t \in \mathcal{T}} \lambda_{ij}^{sd,t}, \quad \begin{matrix} \forall \{i,j\} \in E \\ \forall (s,d) \in D. \end{matrix} \tag{3}$$

- At most one demand can use edge $\{i, j\}$ during slot t

$$\sum_{\forall (s,d) \in D} \lambda_{ij}^{sd,t} \leq \tau_{ij}^{t}, \quad \begin{matrix} \forall \{i,j\} \in E \\ \forall t \in \mathcal{T}. \end{matrix} \tag{4}$$

- Ensure enough capacity exists to support the necessary flow f^{sd} for demand (s, d) on edge $\{i, j\}$ for the length of time that the link is active

$$f^{sd} x_{ij}^{sd} \leq \sum_{t \in \mathcal{T}} \lambda_{ij}^{sd,t} u_{ij}, \quad \begin{matrix} \forall \{i,j\} \in E \\ \forall (s,d) \in D. \end{matrix} \tag{5}$$

- Mark if slot t is used to schedule a demand before a failure

$$\tau_{ij}^{t} \leq s^t, \quad \begin{matrix} \forall \{i,j\} \in E \\ \forall t \in \mathcal{T}. \end{matrix} \tag{6}$$

- Interference constraints:
  - *Binary*: In a time slot, only links that do not interfere with one another can be activated simultaneously

$$\tau_{ij}^{t} + \tau_{uv}^{t} \leq 1 + I_{uv}^{ij}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{u,v\} \in E \\ \{i,j\} \neq \{u,v\},\ \forall t \in \mathcal{T}. \end{matrix} \tag{7a}$$

  - *SINR*: In a time slot, only links that meet the minimum SINR value at the receiver can be activated simultaneously

$$P_{ij} \tau_{ij}^{t} \geq \beta N + \beta \sum_{\substack{\{u,v\} \in E \\ \{i,j\} \neq \{u,v\}}} P_{uj} \tau_{uv}^{t} - M(1 - \tau_{ij}^{t}), \quad \begin{matrix} \forall \{i,j\} \in E \\ \forall t \in \mathcal{T}. \end{matrix} \tag{7b}$$

After a Failure:

- Find a second path for demand (s, d) to be used as the disjoint protection path

$$\sum_{\{i,j\} \in E} y_{ij}^{sd} - \sum_{\{j,i\} \in E} y_{ji}^{sd} = \begin{cases} 1 & \text{if } i = s \\ -1 & \text{if } i = d \\ 0 & \text{otherwise,} \end{cases} \quad \begin{matrix} \forall (s,d) \in D \\ \forall i \in V \end{matrix} \tag{8}$$

- Enforce path disjointness between the primary and protection path for demand (s, d).

- *Edge*-disjoint

$$x_{ij}^{sd} + y_{ij}^{sd} \leq 1, \quad \begin{matrix} \forall (s,d) \in D \\ \forall \{i,j\} \in E. \end{matrix} \tag{9a}$$

- *Node*-disjoint

$$\sum_{j \in V \setminus (s,d)} x_{ij}^{sd} + \sum_{j \in V \setminus (s,d)} y_{ij}^{sd} \leq 1, \quad \begin{matrix} \forall (s,d) \in D \\ \forall i \in V. \end{matrix} \tag{9b}$$

- If after the failure of $\{k,l\}$, the primary path for demand (s,d) did *not* fail (i.e., edge $\{k,l\}$ was not part of the primary path), then that primary path must remain active and use the same schedule as from before the failure. In other words, if edge $\{i,j\}$ was part of the primary path, but the failed edge $\{k,l\}$ was not, then force the same time slot assignment on edge $\{i,j\}$ for after the failure of $\{k,l\}$ that $\{i,j\}$ used before the failure, i.e., $\delta_{ij,kl}^{sd,t} = 1$ if $\lambda_{ij}^{sd,t} = 1$ when $x_{ij}^{sd} = 1$ and $x_{kl}^{sd} = 0$. This can be accomplished by,

$$\lambda_{ij}^{sd,t} + [(x_{ij}^{sd} - x_{kl}^{sd}) - 1] \leq \delta_{ij,kl}^{sd,t}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E \\ \forall t \in \mathcal{T},\ \forall (s,d) \in D. \end{matrix} \tag{10}$$

- If after the failure of edge $\{k,l\}$, the primary path for demand (s,d) *did* fail (i.e., edge $\{k,l\}$ was part of the primary path), schedule the disjoint backup path

$$y_{ij}^{sd} - (1 - x_{kl}^{sd}) \leq \sum_{t \in \mathcal{T}} \delta_{ij,kl}^{sd,t}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E \\ \forall (s,d) \in D. \end{matrix} \tag{11}$$

- At most one demand can use edge $\{i,j\}$ during slot t after the failure of $\{k,l\}$

$$\sum_{\forall (s,d) \in D} \delta_{ij,kl}^{sd,t} \leq \pi_{ij,kl}^{t}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E \\ t \in \mathcal{T}. \end{matrix} \tag{12}$$

- Ensure enough capacity exists after the failure of link $\{k,l\}$ to support the necessary flow f^{sd} on edge $\{i,j\}$ for the length of time that the link is active

$$f^{sd} y_{ij}^{sd} \leq \sum_{t \in \mathcal{T}} \pi_{ij,kl}^{t} u_{ij}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E \\ \forall (s,d) \in D. \end{matrix} \tag{13}$$

- Mark if slot t is used to schedule any demand's disjoint protection path after the failure of $\{k,l\}$

$$\pi_{ij,kl}^{t} \leq s^{t}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E \\ \forall t \in \mathcal{T}. \end{matrix} \tag{14}$$

- Interference constraints (after failure of link $\{k,l\}$):
  - *Binary*: In any given time slot, after the failure of link $\{k,l\}$, only links that do not interfere with one another can be activated simultaneously

$$\pi_{ij,kl}^{t} + \pi_{uv,kl}^{t} \leq 1 + I_{uv}^{ij}, \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{u,v\} \in E \\ \forall \{k,l\} \in E,\ \forall t \in \mathcal{T} \\ \{i,j\} \neq \{u,v\} \neq \{k,l\}. \end{matrix} \tag{15a}$$

  - *SINR*: In any given time slot, after the failure of $\{k,l\}$, only links that meet the minimum SINR value at the receiver can be activated simultaneously

$$P_{ij}\pi_{ij,kl}^{t} \geq \beta N + \beta \sum_{\substack{\{u,v\} \in E \\ \{u,v\} \neq \{i,j\} \neq \{k,l\}}} P_{uj}\pi_{uv,kl}^{t} - M(1 - \pi_{ij,kl}^{t}), \quad \begin{matrix} \forall \{i,j\} \in E,\ \forall \{k,l\} \in E, \{i,j\} \neq \{k,l\},\ \forall t \in \mathcal{T}. \end{matrix} \tag{15b}$$

3.2 Simulation Results for WGP

The Wireless Guaranteed Protection scheme is compared to the traditional 1:1 protection scheme used in wireless networks, an approach that we call wireless 1:1. This method for protection in wireless networks was suggested in [12]. In particular, wireless 1:1 preallocates resources on a disjoint backup path between a source and destination, and switches to that backup path upon a failure in the primary path. The number of time slots to route and schedule the demands without any protection is a lower bound for any solution that includes protection for the same set of demands. Hence, we compare the number of additional time slots needed for protection beyond those that were needed for the case without any protection.

Due to its complexity, an integer linear program can take a long time to run. Because of this, it may not always be possible to obtain an optimal solution, even for small networks; we found this to be the case for WGP. The ILP developed in Section 3.1 jointly optimizes the schedule for before and after a failure. To allow our ILP to run in a reasonable amount of time, we separate the before and after phase, and use a two-step approach: First, we find the routes and schedules for all of the demands "before a failure", then we find the same for "after a failure". While this approach is sub-optimal, our simulations show that the routes and schedules found required only minimal additional time slots for protection beyond the solution without any protection. The "before a failure" phase is the minimum number of time slots to route and schedule the demands without any protection. The minimum number of time slots for the wireless 1:1 scheme is found using an ILP.

For binary interference, the 2-hop interference model is used, which corresponds to the IEEE 802.11 standard [24]. Fifty random graphs were generated with twenty nodes each. Nodes that are within a certain transmission range of one another have a link, and the transmission range is varied to give different desired average node degrees. All links are set to have unit capacity. The node degree is varied from 3.5 to 6.5, and for each graph, twelve source/destination pairs are randomly chosen to be routed concurrently with unit-demand each.

For SINR interference, the standard physical interference model is used, where the power received at node j from node i is based off the transmission power of i and distance between the two nodes. If r_i is node i's transmit power, $d(i,j)$ is the distance from i to j, and α is the path loss exponent, then $P_{ij} = \frac{r_i}{d(i,j)^{\alpha}}$. The SINR formulation has more constraints than the binary case. Hence, to allow for a reasonable runtime using SINR constraints, smaller networks with fewer demands were simulated. Fifty random graphs were generated with fifteen nodes each, and transmission power is held constant for all nodes. Six source/destination pairs are randomly chosen to be routed concurrently. The ambient noise N is set to zero. The reception

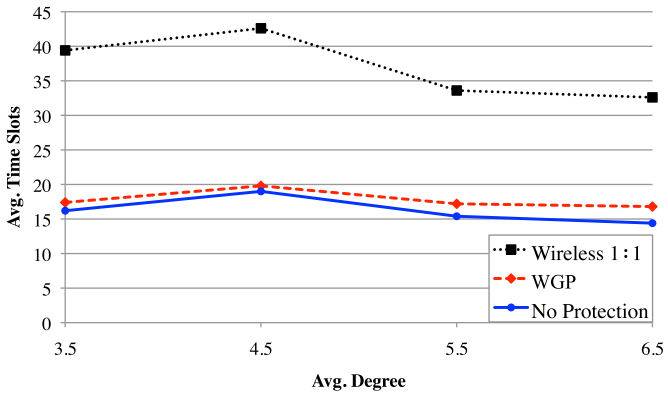


Fig. 2. Binary interference simulation results.

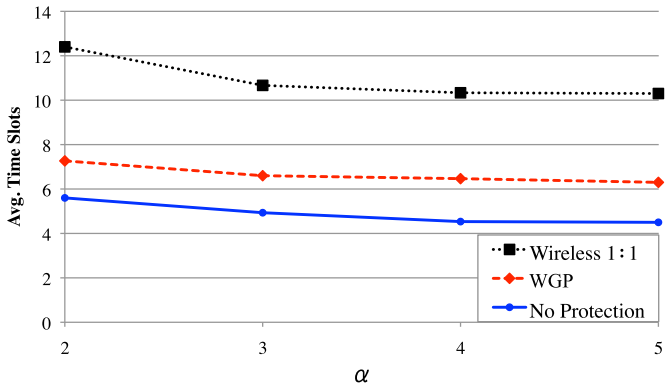


Fig. 3. SINR interference simulation results.

threshold β is set to 4.5, and α is varied from 2 to 5. For both interference models, we simulate only the edge-disjoint case.

The results for binary interference are plotted in Fig. 2. On average, WGP used 94 percent fewer time slots to provide the same level of resiliency as that of wireless 1:1. In fact, for 50 percent of the cases tested, WGP needed no additional time slots beyond what was required to route and schedule the demands without protection. On average, WGP needed only 8 percent more time slots beyond that of the no protection case, while wireless 1:1 needed 128 percent additional time slots. For the most part, the same time slots that were used to schedule the primary paths can be reused to schedule the disjoint backup paths. The results for SINR interference are plotted in Fig. 3. On average, WGP used 82 percent fewer time slots for protection than wireless 1:1. No additional protection time slots were needed for 18 percent of cases. Furthermore, the savings remain relatively constant as α increases.

4 Algorithms for Providing Wireless Protection

In the previous section, an integer linear program was presented to find the minimum length schedule for Wireless Guaranteed Protection. An ILP is not a computationally efficient method of finding a solution; in fact, the ILP in Section 3 needed to be split into two parts to allow it to run in a reasonable amount of time. In this section, we develop a time-efficient algorithm to solve WGP for both general binary and SINR interference constraints. As was demonstrated in Section 3, an optimal solution to WGP is NP-hard, even to approximate. To solve WGP, we utilize a dynamic approach that will route and schedule each demand one-at-a-time, where each demand is scheduled such that it does not interfere with previously scheduled connections. In Theorem 2, we show that even when demands are routed one-at-a-time, finding the minimum number of time slots to route and schedule any individual demand is NP-hard.

Theorem 2. *When demands are routed and scheduled one-at-a-time with disjoint path protection, the minimum number of time slots for any individual demand is NP-hard to determine using either binary or SINR interference constraints when accounting for the time slots that are currently in use.*

To prove Theorem 2, a reduction from the Dynamic Shared-Path-Protected Lightpath-Provisioning Problem (DSPLP) [4] is performed. The proof can be found in Appendix B.

Since it is NP-hard to determine the minimum number of time slots to route and schedule any individual demand, the algorithm will work in the following fashion: For each demand, a route and schedule is first found for the primary path, and then a route and schedule is found for the disjoint protection path. In Section 4.1, an algorithm to find an interference-free path for both the binary and SINR interference model is presented. This path is found as follows: We first find a path that is of low-interference (a metric that we define later), and then determine an interference-free schedule for that path. This path algorithm is then used as a subroutine to efficiently solve WGP, which is presented in Section 4.2.

4.1 Interference-Free Path with a Minimal Length Schedule

In this section, an algorithm is developed to find an interference-free path, which will then be used to construct the disjoint path protection algorithm in the following section. We assume connections already exist in the network, and are scheduled using the set of $\mathcal{T}$ time slots. We desire to set up a new path from s to d. We take a two-step approach: First, find a path that is of "low-interference", and then find a minimal length schedule for this path. We call this algorithm `feasible_path`.

4.1.1 Low-Interference Path

Given that a set of connections already exist using the set of $\mathcal{T}$ time slots across the set of edges E, each edge can be assigned a value according to its general "interference load", which we define to be the set of time slots that cannot be used for some particular edge. These time slots may not be available because either that edge uses them, or some interfering edge uses them. For edge $\{i, j\}$, we label the set of unavailable time slots τ_{ij}. If an edge that is heavily loaded (few available time slots) is used for a connection, then that may prevent some future connection from being able to find an interference-free path without the use of additional time slots. To find paths that are of low-interference, we assign a cost to each edge that is equal to its interference load: $c_{ij} = |\tau_{ij}|$. We then find a shortest path from s to d with respect to these edge-costs, giving preference to edges that are not heavily loaded.

We build the set τ_{ij} for edge $\{i,j\}$ in the following manner. Define the set of time slots that are currently assigned to edge $\{i,j\}$ as t_{ij}. For binary interference, label the set of edges that $\{i,j\}$ interferes with as γ_{ij}. The set of time slots not available for use on $\{i,j\}$ are the ones currently assigned to $\{i,j\}$ and to the set of edges γ_{ij}: $\tau_{ij} = t_{ij} \cup_{\{k,l\}\in\gamma_{ij}} t_{kl}$. For SINR interference, determining the set of unavailable time slots is not as straightforward; interference between edges depends on the power received at some node coming from all of the other transmitting nodes. For SINR, we simply define τ_{ij} to be the set of time slots in use on that edge: $\tau_{ij} = t_{ij}$.

The complexity of calculating a low-interference path is as follows. To calculate edge costs for binary interference, each edge is assigned a value based on the utilization of interfering edges. Since each edge may potentially interfere, each edge needs to be considered, and the time slots used on each edge need to enumerated. Thus the complexity for calculating edge costs is $O(|\mathcal{T}||E|^2)$, where $\mathcal{T}$ is the set of time slots in use and E is the set of edges in the graph. For SINR interference, interfering edges are not taken into consideration when calculating edge costs, thus having lower complexity than the binary interference case. After the edge costs are assigned, a shortest path algorithm is utilized to find the low-interference path, which has complexity of $O(|V|^2)$, where V is the set of vertices in the graph [33]. Since $|E| \geq |V|$, the worst case complexity for finding a low-interference path is $O(|\mathcal{T}||E|^2)$.

4.1.2 Minimal Length Schedule for a Path

Once a low-interference path has been found, we want to find a minimal length schedule for it.

Binary Interference. We construct a conflict graph G^c, which is built as follows: For each edge $\{i,j\}$ in the original graph G, a node v_{ij} is added in G^c. If two edges $\{i,j\}$ and $\{k,l\}$ in G cannot be activated simultaneously because they interfere with one another, then in G^c, an edge is added between nodes v_{ij} and v_{kl} in G^c [15]. Any independent set[1] of G^c are a set of edges in the original graph that can be activated simultaneously. Any feasible coloring[2] of the nodes of G^c is a feasible schedule of link activations. Label the set of edges of the path as P. We construct G^c using only the set of edges P: Add node v_{ij} to G^c for each edge in P, and add an edge between v_{ij} and v_{kl} if edges $\{i,j\}$ and $\{k,l\}$ cannot be active simultaneously.

We wish to find a minimum node-coloring of G^c, which will be a minimum-length schedule for P. The minimum node-coloring problem is NP-hard to solve [34]. For our problem, we have a restriction that not all colors are available for all nodes: The set of colors not available for node v_{ij} is the set of time slots that edge $\{i,j\}$ cannot use: τ_{ij}. We note that this restricted node-coloring problem remains NP-hard. A valid instance of the restricted problem is to have $\tau_{ij} = \emptyset$, $\forall\{i,j\}$, which is simply the original NP-hard node-coloring problem. To find a solution, we use the Welsh-Powell algorithm that colors the nodes (assigns time slots) in a greedy fashion, starting with nodes that have highest degree [34].

1. An independent set is a set of nodes where no two nodes are the end points of the same edge.

2. Each node is assigned a color such that all nodes of one color form an independent set.

The complexity of finding a minimal length schedule for a path using binary interference is as follows. To create a conflict graph, each edge of the original graph becomes a node in the conflict graph. Then, each node in the conflict graph forms an edge with another node if the two edges in the original graph interfered with one another. Since an edge in the original graph may interfere with all other edges, up to $|E|$ edges may be formed from any particular node in the conflict graph. Hence, the complexity of creating a conflict graph is $O(|E|^2)$. To find the minimum node-coloring, we utilize the Welsh-Powell greedy coloring algorithm, which has complexity $O(v^2)$ for a graph with v vertices [34]. Since the conflict graph has $|E|$ vertices, the node-coloring operation takes $O(|E|^2)$. The complexity to create the conflict graph, and find a minimum node-coloring is $O(|E|^2)$, and the complexity of finding a low-interference path is $O(|\mathcal{T}||E|^2)$. Hence, the worst case complexity of `feasible_path` for binary interference is $O(|\mathcal{T}||E|^2)$.

SINR Interference. Label V^t as the set of nodes currently transmitting during time slot t. Node j can only receive a transmission from i if the power received from node i is above some factor of the power being received from all other currently transmitting nodes during time slot t, i.e., $P_{ij} \geq \beta(N + \sum_{v\in V^t\setminus i} P_{vj})$. We wish to determine if some edge $\{k,l\}$ can be assigned time slot t. Edge $\{k,l\}$ can use time slot t if the following two conditions are met:

1) The power heard at node l is sufficiently greater than the other nodes that are currently transmitting in time slot t: $P_{kl} \geq \beta(N + \sum_{v\in V^t} P_{vl})$.
2) Label E^t as the set of links currently transmitting during time slot t. Using edge $\{k,l\}$ does not interfere with some existing transmission on link $\{i,j\} \in E^t$: $P_{ij} \geq \beta(N + \sum_{v\in V^t\setminus i} P_{vj} + P_{kj})$, $\forall\{i,j\} \in E^t$.

We wish to schedule path P. For each edge in P, an available time slot is found that meets the above two conditions. Similar to the approach taken in [27], time slots are assigned in a greedy manner, starting with the edges that cause the most interference with other edges.

The complexity of finding a minimal length schedule for a path using SINR interference is as follows. The path P may have up to $|E|$ edges, and for each edge in P we check the total power to and from all other receiving nodes and emitting nodes, respectively. This has a complexity of $O(|E||V|)$. The complexity to find a low-interference path is $O(|\mathcal{T}||E|^2)$, which is worse than the $O(|E||V|)$. Hence, the worst case complexity of `feasible_path` for SINR interference is $O(|\mathcal{T}||E|^2)$.

4.2 Wireless Guaranteed Protection

In Section 4.1, an algorithm `feasible_path` was presented that finds a path and schedule between two nodes that takes into account other scheduled connections in the network. We use `feasible_path` as a subroutine to construct an algorithm for WGP. We label the algorithm presented in this section `WGP_alg`. We present the algorithm for the edge-disjoint protection case, but it can be easily modified to work for the node-disjoint case as well. Since the subroutine `feasible_path` finds a path with respect to either binary or SINR interference

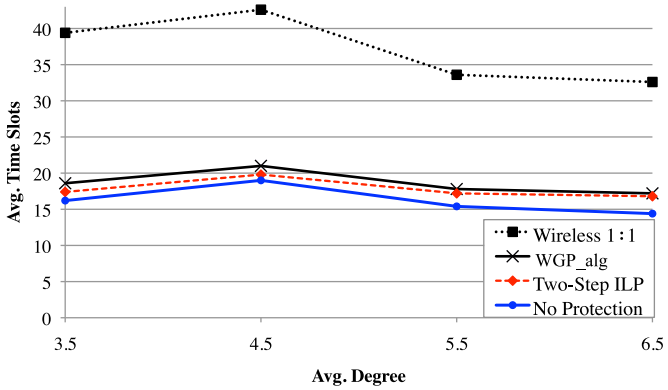


Fig. 4. Binary interference algorithm simulation.

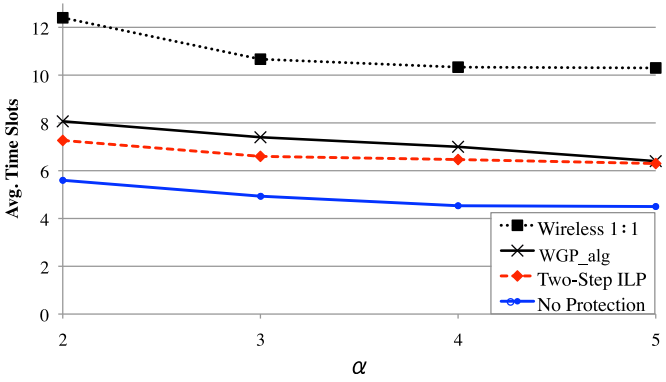


Fig. 5. SINR interference algorithm simulation.

constraints, WGP_alg is agnostic to the interference constraints used.

The mechanism for disjoint path protection in wireless networks is as follows. Each demand will have a primary and backup path, as well as an interference-free schedule for those paths, and after the failure of some edge, a demand whose primary path fails will switch to its disjoint backup path and schedule. To minimize network disruption after a failure, if a demand's primary path did not fail, it will continue to use its pre-failure primary path and schedule. If a primary path does fail, the time slots that were used to schedule that path are no longer needed, and can be reused for the protection path.

We consider some incoming demand between nodes s and d, with the network already having some set of scheduled connections using the set of time slots $\mathcal{T}$. As defined in Section 4.1.1, τ_{ij} is the set of time slots that cannot be used to schedule edge $\{i,j\}$, which we called the "interference load". We call the set of interference loads for each edge the *interference set*, and we label it $\Gamma = \{\tau_{ij} | \{i,j\} \in E\}$. Because of the different sets of paths used, the interference set can be different before a failure and after any particular failure. For the existing scheduled connections, the interference set before any failure (only the primary paths) is labeled Γ, and is labeled Γ_{kl} for after the failure of edge $\{k,l\}$. The interference set Γ_{kl} reflects the schedules of all the paths that are currently used in the event of the failure of $\{k,l\}$, which includes the backup paths for demands that fail, as well as the primary paths for the demands that did not fail.

The algorithm for Wireless Guaranteed Protection (WGP_alg) is as follows. First, using the interference set Γ, a path and its corresponding schedule is found between s and d using feasible_path. This will be the primary path, and we label its set of edges as P. Next, we find the disjoint backup path. We construct a new graph G^F that does not have the set of edges P; any path between s and d in G^F will be disjoint to P. We consider the possible failure of any edge in the primary path. Upon the failure of edge $\{k,l\} \in P$, demands that did not fail must continue to use their pre-failure path and schedule, and demands that did fail switch to their backup path and schedule. After the failure of an edge in P, the edges of that path no longer supports any flow, and the time slots used on those edges become available for protection. We form a new interference set that contains the information of all the possible paths used after the failure of any edge in the primary path: $\Gamma^F = \cup_{\{k,l\} \in P} \Gamma_{kl}$. Since Γ^F does not contain any scheduling information regarding P, the time slots used to schedule P can be reused to schedule the disjoint backup path. Using graph G^F and interference set Γ^F, a path is found between s and d using feasible_path, which is the disjoint backup path.

The complexity of WGP_alg is as follows. To calculate the interference set Γ_{kl}, the set of time slots in use on each edge after the failure of $\{k,l\}$ is examined. If the set of the timeslots in use is $\mathcal{T}$, then calculating Γ_{kl} takes $O(|\mathcal{T}||E|)$, and finding the entire set Γ^F takes $O(|\mathcal{T}||E|^2)$. In addition to calculating the set Γ^F, two runs of feasible_path are used, where the complexity of feasible_path is $O(|\mathcal{T}||E|^2)$. Hence, the worst case complexity of WGP_alg is $O(|\mathcal{T}||E|^2)$.

To demonstrate the performance of WGP_alg, we simulate the algorithm using the same parameters as the simulation for the ILP in Section 3.2. For WGP_alg, the demands are randomly ordered, and a route and schedule is found for each demand one-at-a-time. The algorithm is compared to the wireless 1:1 scheme and the two-step ILP, both of which were described in Section 3.2. Fig. 4 shows the simulation results for binary interference, and Fig. 5 shows the results for SINR interference. For binary interference, on average, WGP_alg performed within 4 percent of the two-step ILP, and required 88 percent fewer protection time slots than wireless 1:1. For SINR interference, on average, WGP_alg performed within 8 percent of the two-step ILP, and required 78 percent fewer protection time slots than wireless 1:1.

Finally, we compare the run-time of the two-step ILP to the developed algorithm WGP_alg. The purpose of our developing an alternative solution to the ILP formulation is to achieve a significantly faster run-time, and this is in fact what we find. The two-step ILP was solved using IBM ILOG CPLEX 12.1.0, and WGP_alg was implemented using Perl. In Fig. 6, the time to run WGP_alg and the two-step ILP for binary interference constraints is plotted with respect to the average node degree.[3] As the node degree increases, we see that the run-time of the ILP formulation has a dramatic increase, going from an average of around 1,800 seconds for node degree of 3.5 to an average of over 10,000 seconds for a node degree of 6.5. One reason for this

3. We note that only the results for binary interference constraints are shown, but similar results were seen for SINR constraints.

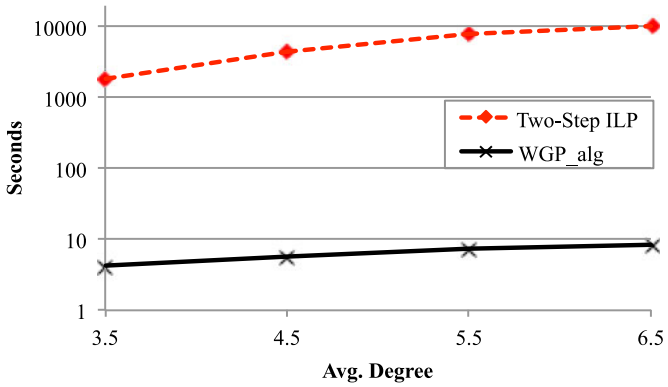


Fig. 6. Run-time of the developed algorithm `WGP_alg` versus the two-step ILP formulation for binary interference constraints.

dramatic increase in run-time is because as the density of the network increases, the number of edges increases, which results in a significant increase in the number of constraints for the ILP. In contrast, the algorithm has an average run-time of 4.3 seconds at node degree of 3.5 and goes only to a run-time of 8.7 seconds for node degree of 6.5. This represents a savings of over three orders of magnitude with respect to run-time. Furthermore, the scaling behavior of the ILP is worse than that of the algorithm. The ILP has a five-fold increase in run-time when going from node degree of 3.5 to 6.5, while the algorithm has less than a two-fold increase in run-time.

5 Conclusion

In this paper, the problem of Wireless Guaranteed Protection for networks subject to interference constraints was examined. The motivation is to provide protection that is similar to that of wired networks, but designed for wireless networks. Solutions using both general binary and SINR interference constraints were developed to allow our formulation to be applied to almost any interference model. Our protection scheme takes advantage of the interference in wireless networks for greater resource efficiency: Resources that are freed after a failure in the network can be reused for protection from that failure. We demonstrated that WGP is NP-hard, even to approximate. We formulated an ILP to solve WGP, giving solutions that used 87 percent fewer protection resources on average than the wired disjoint path scheme in wireless networks. For the case of 2-hop interference, which approximates to the IEEE 802.11 standard, our protection scheme requires only 8 percent more resources on average than providing no protection whatsoever. We then developed a time-efficient algorithm for WGP that performed almost as well as the two-step ILP on average. The algorithm has a run-time that is up to three orders of magnitude faster than the two-step ILP. A future direction for our work is to adapt the schemes developed in this paper to a distributed setting.

Appendix A
Proof of Theorem 1

To prove the NP-hardness and non-approximability of Wireless Guaranteed Protection, we reduce from the problem of determining the chromatic number of a graph with n nodes [34]. Given a graph $G^c = (V, E)$, the chromatic number of the graph is the minimum number of colors needed to color each node of the graph such that any two nodes connected by an edge do not have the same color. A graph coloring of n is always possible by assigning each node its own color. For all $\epsilon > 0$, it is NP-hard to approximate the chromatic number to within $n^{1-\epsilon}$ [34].

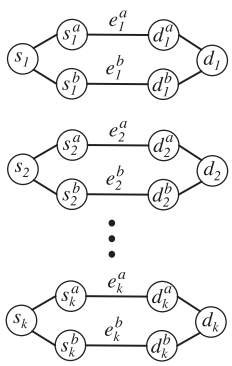


Fig. 7. Communication network G for proof of Theorem 1.

We perform the following polynomial time reduction of chromatic number to WGP. We construct a communication network G as follows (shown in Fig. 7). For node v_i in G^c, associate a demand from s_i to d_i in G. A pair of disjoint paths will exist from s_i to d_i, $\forall i$. One path traverses nodes s_i^a and d_i^a, which are connected by edge e_i^a, and the other uses nodes s_i^b and d_i^b, connected by edge e_i^b. Since there exists only two possible paths from s_i to d_i to satisfy the ith demand, and a protection path is guaranteed to exist after any single link failure, one of the two paths will be the primary path, and the other is the backup path. We use e_i^* to denote either e_i^a or e_i^b, and s_i^* for either s_i^a or s_i^b.

Next, we assign interferences such that finding the minimum number of time slots for WGP in G will simultaneously find the chromatic number of G^c. First, we consider binary interference. In the original graph G^c, if there exists an edge between nodes i and j, then the edges in G associated with those nodes (e_i^* and e_j^*) interfere with one another, i.e., cannot be active at the same time. We note that it does not matter if e_i^a interferes with e_i^b since they will never be active simultaneously (one will be active before a failure, and the other will be active after). The set of edges in G associated with the nodes in G^c are the only ones that cause interference. All other edges can be activated with any other edge.

Second, we consider SINR interference. In the original graph, G^c, if there exists an edge $\{i, j\}$, then we assign transmission powers such that node d_j^* cannot hear s_j^* if s_i^* is transmitting, but can otherwise. This interference scheme can be accomplished using binary entries in the power matrix, and setting $\beta > 1$ and $N = 0$. Recall that P_{uv} is the power received at v when u is transmitting, and for a transmission to be successful from node u to v, the following condition must hold (when $N = 0$): $P_{uv} \geq \beta \sum_{v' \in V'} P_{v'v}$, where V' is the set of nodes that are currently transmitting, excluding u. If in G^c, there exists an edge $\{i, j\}$, then nodes i and j cannot have the same color. For this case, in G, when s_i^* is transmitting, we set $P_{s_i^* d_j^*}$ to 1, and d_j^* can no longer receive from s_j^*. Similarly, when s_j^* is transmitting, we set $P_{s_j^* d_i^*}$ to 1. Alternatively, if there did not exist an edge $\{i, j\}$ in G^c, then

nodes i and j can potentially share the same color. For this case, in G, we set $P_{s_i^* d_j^*}$ (and $P_{s_j^* d_i^*}$) to 0, and when s_i^* is transmitting, d_j^* can still receive from s_j^*. Again, it does not matter if s_i^a interferes with d_i^b, or if s_i^b interferes with d_i^a, since edges e_i^a and e_i^b will never be active at the same time. All other nodes (the source and destination nodes s_i and d_i, $\forall i$) can receive without interference and can transmit without interfering.

The upper bound for the number of time slots to solve WGP in G is n: One time slot for each demand. The minimum number of time slots will be the chromatic number of G^c. If there existed a polynomial time algorithm to approximate WGP, then that algorithm can be used to approximate the chromatic number of a graph in polynomial time.

APPENDIX B
PROOF OF THEOREM 2

To prove Theorem 2, we reduce the Dynamic Shared-Path-Protected Lightpath-Provisioning Problem [4] to Wireless Guaranteed Protection. DSPLP finds disjoint paths for demands one-at-a-time, and assigns wavelengths of light for the edges of those paths. Each edge of a path can use a different wavelength for communication. The restrictions on wavelength assignments are: No two primary paths can share a wavelength on an edge; a wavelength can be shared for protection on an edge only if the two demands have failure disjoint primary paths, i.e., only one will fail at a time. In [4], the authors show that finding an eligible pair of primary and backup paths for an incoming demand using the set of available wavelengths is NP-complete. Hence, if determining if there exists a primary and backup path using some set of wavelengths is NP-complete, it is NP-hard to find the minimum number of wavelengths to do so.

WGP finds disjoint paths for demands one-at-a-time, and assigns time slots to the different edges of those paths so that transmissions can occur without interference. Removing interference from the network will make WGP equivalent to DSPLP. A valid instance of WGP that effectively has no interference is as follows: For binary interference, allow all edges to be active simultaneously; for SINR interference, set the reception threshold to zero: $\beta = 0$. If there exists a polynomial time algorithm to find the minimum number of time slots to route and schedule a demand in WGP, then there exists a polynomial time algorithm to find the minimum number of wavelengths for DSPLP.

ACKNOWLEDGMENTS

This work was supported by US National Science Foundation grants CNS-1116209 and CNS-0830961, by DTRA grant HDTRA-09-1-005.

REFERENCES

[1] M. D. Williams, *Broadband for Africa: Developing Backbone Communications Networks*. Washington, DC, USA: World Bank Publications, 2010.

[2] J. P. Sterbenz, et al., "Survivable mobile wireless networks: Issues, challenges, and research directions," in *Proc. 1st ACM Workshop Wireless Security*, 2002, pp. 31–40.

[3] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks," *J. Lightwave Technol.*, vol. 21, no. 4, 2003, Art. no. 870.

[4] C. Ou, J. Zhang, H. Zang, L. Sahasrabuddhe, and B. Mukherjee, "New and improved approaches for shared-path protection in WDM mesh networks," *J. Lightwave Technol.*, vol. 22, no. 5, pp. 1223–1232, May 2004.

[5] B. Mukherjee, "WDM optical communication networks: Progress and challenges," *IEEE J. Selected Areas Commun.*, vol. 18, no. 10, pp. 1810–1824, Oct. 2000.

[6] W. Yao and B. Ramamurthy, "Survivable traffic grooming with path protection at the connection level in WDM mesh networks," *J. Lightwave Technol.*, vol. 23, no. 10, pp. 2846–2853, Oct. 2005.

[7] A. Saleh and J. Simmons, "Evolution toward the next-generation core optical network," *J. Lightwave Technol.*, vol. 24, no. 9, pp. 3303–3321, Sep. 2006.

[8] A. Koster and X. Muñoz, *Graphs and Algorithms in Communication Networks: Studies in Broadband, Optical, Wireless and ad hoc Networks*. Berlin, Germany: Springer, 2009.

[9] W. Grover, *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET, and ATM Networking*. Englewood Cliffs, NJ, USA: Prentice Hall, 2004.

[10] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 5, no. 4, pp. 11–25, 2001.

[11] N. Ahmed, S. Kanhere, and S. Jha, "The holes problem in wireless sensor networks: A survey," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 9, no. 2, pp. 4–18, 2005.

[12] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless networks: The joint routing and scheduling problem," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw.*, 2003, pp. 42–54.

[13] G. Kuperman and E. Modiano, "Providing protection in multi-hop wireless networks," in *Proc. IEEE INFOCOM*, 2013, pp. 926–934.

[14] G. Kuperman and E. Modiano, "Disjoint path protection in multi-hop wireless networks with interference constraints," in *Proc. IEEE Global Commun. Conf.*, 2014, pp. 4472–4477.

[15] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," *Wirel. Netw.*, vol. 11, no. 4, pp. 471–487, 2005.

[16] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. 11th Annu. Int. Conf. Mobile Comput. Netw.*, 2005, pp. 58–72.

[17] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 910–917, Sep. 1988.

[18] W. Wang, Y. Wang, X. Li, W. Song, and O. Frieder, "Efficient interference-aware TDMA link scheduling for static wireless networks," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 262–273.

[19] J. Zhang, H. Wu, Q. Zhang, and B. Li, "Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks," in *Proc. 2nd Int. Conf. Broadband Netw.*, 2005, pp. 631–640.

[20] R. Gupta, J. Musacchio, and J. Walrand, "Sufficient rate constraints for QoS flows in ad-hoc networks," *Ad Hoc Netw.*, vol. 5, no. 4, pp. 429–443, 2007.

[21] X. Lin and S. Rasool, "A distributed joint channel-assignment, scheduling and routing algorithm for multi-channel ad-hoc wireless networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 1118–1126.

[22] V. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan, "Algorithmic aspects of capacity in wireless networks," in *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 33, no. 1, pp. 133–144, 2005.

[23] A. P. Subramanian, H. Gupta, S. R. Das, and J. Cao, "Minimum interference channel assignment in multiradio wireless mesh networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 12, pp. 1459–1473, Dec. 2008.

[24] G. Sharma, R. Mazumdar, and N. Shroff, "On the complexity of scheduling in wireless networks," in *Proc. 12th Annu. Int'l Conf. Mobile Comput. Netw.*, 2006, pp. 227–238.

[25] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[26] U. C. Kozat, I. Koutsopoulos, and L. Tassiulas, "A framework for cross-layer design of energy-efficient communication with QoS provisioning in multi-hop wireless networks," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. and Commun. Soc.*, 2004, pp. 1446–1456.

[27] G. Brar, D. M. Blough, and P. Santi, "Computationally efficient scheduling with the physical interference model for throughput improvement in wireless mesh networks," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw.*, 2006, pp. 2–13.
[28] O. Goussevskaia, R. Wattenhofer, M. M. Halldórsson, and E. Welzl, "Capacity of arbitrary wireless networks," in *Proc. IEEE INFOCOM*, 2009, pp. 1872–1880.
[29] G. Sharma, C. Joo, N. Shroff, and R. Mazumdar, "Joint congestion control and distributed scheduling for throughput guarantees in wireless networks," *ACM Trans. Model. Comput. Simulation*, vol. 21, no. 1, 2010, Art. no. 5.
[30] T. Bauschert, C. Büsing, F. D'Andreagiovanni, A. M. Koster, M. Kutschka, and U. Steglich, "Network planning under demand uncertainty with robust optimization," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 178–185, Feb. 2014.
[31] F. D'Andreagiovanni, C. Mannino, and A. Sassano, "Gub covers and power-indexed formulations for wireless network design," *Management Science*, vol. 59, no. 1, pp. 142–156, 2013.
[32] F. D'Andreagiovanni, C. Mannino, and A. Sassano, "Negative cycle separation in wireless network design," in *Network Optimization*. Berlin, Germany: Springer, 2011, pp. 51–56.
[33] T. H. Cormen, *Introduction to Algorithms*. Cambridge, MA, USA: MIT press, 2009.
[34] D. Zuckerman, "Linear degree extractors and the inapproximability of max clique and chromatic number," in *Proc. 38th Annu. ACM Symp. Theory Comput.*, 2006, pp. 681–690.

Greg Kuperman received the BSE and MSE degrees in electrical engineering from the University of Pennsylvania, and the PhD degree in communications and networking from the Massachusetts Institute of Technology. He is a member of the technical staff in the Tactical Networks Group at the MIT Lincoln Laboratory.

Eytan Modiano received the BS degree in electrical engineering and computer science from the University of Connecticut, Storrs, in 1986 and the MS and PhD degrees, both in electrical engineering from the University of Maryland, College Park, Maryland, in 1989 and 1992, respectively. He was a naval research laboratory fellow between 1987 and 1992 and a national research council post doctoral fellow during 1992-1993. Between 1993 and 1999, he was with MIT Lincoln Laboratory. Since 1999, he has been on the faculty at MIT, where he is a professor in the Department of Aeronautics and Astronautics and the Laboratory for Information and Decision Systems (LIDS). His research is on communication networks and protocols with emphasis on satellite, wireless, and optical networks. He is the corecipient of the MobHoc 2016 Best Paper Award, the Wiopt 2013 Best Paper Award, and the Sigmetrics 2006 Best Paper Award. He is an editor-at-large for the *IEEE/ACM Transactions on Networking* and served as associate editor for the *IEEE Transactions on Information Theory*, and the *IEEE/ACM Transactions on Networking*. He was the Technical Program co-chair for IEEE Wiopt 2006, IEEE Infocom 2007, ACM MobiHoc 2007, and DRCN 2015. He is a fellow of the IEEE and an associate fellow of the AIAA, and served on the IEEE Fellows committee.