Mobile Netw Appl (2008) 13:198–211
DOI 10.1007/s11036-008-0034-8

Gateway Placement for Throughput Optimization in Wireless Mesh Networks

Fan Li · Yu Wang · Xiang-Yang Li · Ashraf Nusairat · Yanwei Wu

Published online: 8 March 2008

Abstract In this paper, we address the problem of gateway placement for throughput optimization in multi-hop wireless mesh networks. Assume that each mesh node in the mesh network has a traffic demand. Given the number of gateways to be deployed (denoted by k) and the interference model in the network, we study where to place exactly k gateways in the mesh network such that the total throughput is maximized while it also ensures a certain fairness among all mesh nodes. We propose a novel grid-based gateway deployment method using a cross-layer throughput optimization, and prove that the achieved throughput by our method is a constant times of the optimal. Simulation results demonstrate that our method can effectively exploit the available resources and perform much better than random and fixed deployment methods. In addition, the proposed method can also be extended to work with multi-channel and multi-radio mesh networks under different interference models.

Keywords gateway deployment · throughput optimization · link scheduling · wireless mesh networks

1 Introduction

Wireless mesh network (WMN) [1] draws lots of attention in recent years due to its various potential applications, such as broadband home networking, community and neighborhood networks, and enterprise networking. It has also been used as the last mile solution for extending the Internet connectivity for mobile nodes. Many cities and wireless companies have already deployed mesh networks around the world. For example, in Cambridge, UK, on the 3rd June 2006, mesh network was used at the "Strawberry Fair" to run mobile live television, radio and internet services to an estimated 80,000 people. AWA, the Spanish operator of Wireless LAN networks, will roll out commercial WLAN and mesh networks for voice and data services. Several companies such as MeshDynamics have recently announced the availability of multi-hop multi-radio mesh network technology. These networks behave almost like wired networks since they have infrequent topology changes, limited node failures, etc. For wireless mesh networks, the aggregated traffic load of each routing node changes infrequently also. A unique characteristic of wireless networks is that the communication channels are shared by the wireless terminals. Thus,

F. Li · Y. Wang (✉)
Department of Computer Science,
University of North Carolina, Charlotte, NC, USA
e-mail: yu.wang@uncc.edu

F. Li
e-mail: fli@uncc.edu

X.-Y. Li · A. Nusairat · Y. Wu
Department of Computer Science,
Illinois Institute of Technology, Chicago, IL, USA

X.-Y. Li
e-mail: xli@cs.iit.edu

A. Nusairat
e-mail: nusaash@iit.edu

Y. Wu
e-mail: ywu24@iit.edu

one of the major problems facing wireless networks is the reduction of capacity due to interference caused by simultaneous transmissions. Using multiple channels and multiple radios can alleviate but not eliminate the interference.

Wireless mesh networks consist of two types of nodes: mesh routers and mesh clients. Mesh routers form an infrastructure (called mesh backbone) for mesh clients that connect to them. The mesh backbone can be built using various types of radio technologies. The mesh routers form a mesh of self-configuring, self-healing links among themselves. Compared with conventional wireless routers, mesh routers can achieve the same coverage with much lower transmission power through multi-hop communication. To connect the mesh network to the Internet, gateway devices are needed. Usually, in mesh networks some mesh routers have the gateway functionality which can provide the connectivity to the Internet. The common network infrastructure for mesh networks is illustrated in Fig. 1, where dash and solid lines indicate wireless and wired links respectively. We do not include the mesh clients in the figure, since this paper focuses on the design of the mesh backbone only. Hereafter, we will call the mesh routers without gateway functionality *mesh nodes* or just mesh routers, and call the mesh routers with gateway functionality *gateway nodes* to distinguish them from mesh nodes.

In this paper, we study how to design the mesh backbone to optimize the network throughput under the interference. More specifically, given the mesh backbone and the number of gateway devices, we investigate where to place the gateway devices in the mesh backbone in order to achieve optimal throughput. The application scenario of this gateway deployment problem for a community network is as follows. The mesh routers

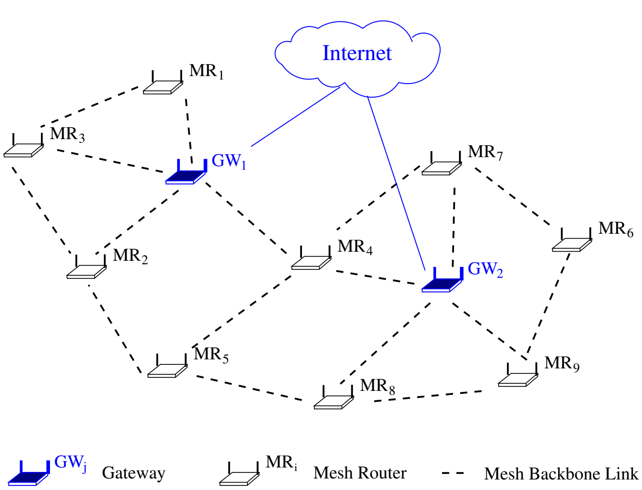


Figure 1 The network infrastructure of wireless mesh network

are placed on the roof of houses in a neighborhood, which serve as access points for users inside the homes and along the roads. All these mesh routers are fixed and form the mesh network. The mesh service provider needs to decide where to put the gateway devices to connect the mesh network to the Internet. Since different gateway deployment causes different mesh backbone topology and affects the network throughput, it is important to find optimal gateway deployment to maximize the throughput.

Optimizing the throughput has been studied in wireless networks. Gupta and Kumar [2] studied the asymptotic capacity of a multi-hop wireless networks. Recently, several papers [3, 4] further investigated the capacity of wireless networks under different models. Kyasanur and Vaidya [5] studied the capacity region on random multi-hop multi-radio multi-channel wireless networks. On the other aspect, several papers [6–9] recently researched on how to satisfy a certain traffic demand vector from all wireless nodes by a joint routing, link scheduling, and channel assignment under certain wireless interference models. Kodialam and Nandagopal [6] considered the problem of jointly routing the flows and scheduling transmissions to achieve a given rate using the protocol interference model in a single channel wireless network. In [7], they extended their work to the multi-radio multi-channel networks. Alicherry et al. [8] presented a linear programming (LP) based method to jointly perform multi-path routing, link scheduling, and static channel assignment for throughput optimization in multi-radio multi-channel wireless networks. Li et al. [9] studied the similar problem with more complex interference models (non-uniform interference range) and dynamic channel assignment schemes. All these studies either focused on the capacity of pure multi-hop mesh networks without gateways or assumed that the positions of mesh nodes and gateway nodes are fixed and given. In this paper, we consider the deployment of gateway nodes which affects the network throughput and capacity.

The deployment schemes of access points in WLAN has been studied [10–14] as well. However, most of the work focused on the guarantee of the coverage or how to provide better coverage using minimum number of access points. For example, Kouhbor et al. [14] studied how to find the optimal number of access points and their locations for WLAN in an environment that includes obstacles. Notice that WLAN is different with WMN since WLAN only supports single-hop wireless communication while WMN is a multi-hop network. For multi-hop networks or hybrid networks, until recently there is only a few studies on deployment of relay nodes or access points. Pabst et al. [15] showed that deployment of fixed relay nodes can enhance capacity in hybrid cellular networks. Fong et al. [16] also studied some fixed broadband wireless access deployment schemes to increase the network capacity.

The work closest to ours is the pioneering work in [24]. Chandra et al. [24] developed algorithms to place internet gateways (called ITAPs there) in multi-hop wireless network to minimize the number of gateways while satisfying users' bandwidth requirements. They formed the gateway placement problems as linear programs and presented several greedy-based approximation algorithms. The major differences between their work and ours are: (1) they used coarse-grained interference model that estimates a relation between throughput and wireless interference, while in this paper we adopt fine-grained interference model based on conflict graph; (2) their goal of deployment is to minimize the number of gateways, while ours is to maximize the throughput using fixed number of gateways; and (3) they considered that the set of finite possible gateway locations is given, while we consider all locations in a region which leads to infinite possible locations. To the best of our knowledge, there is no previous study on how to deployment gateways in wireless mesh networks to maximize the throughput.

The rest of the paper is organized as follows. In Section 2, we present our network model and interference model. We then mathematically formulate the throughput optimization problem for a fixed mesh network and give a greedy scheduling algorithm which can achieve constant times of the optimal throughput in Section 3. In Section 4, we present an efficient grid-based gateway deployment scheme for throughput optimization and prove that the achieved throughput by our method is a constant times of the optimal. Our simulation results are presented in Section 5. We discuss possible extensions of our proposed scheme in Section 6. Section 7 concludes our paper.

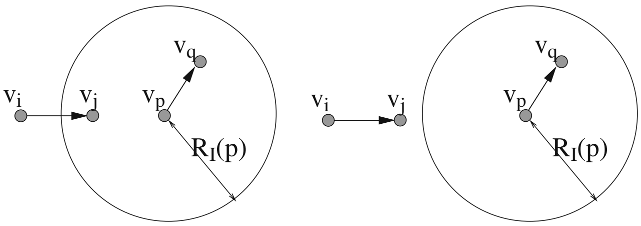
2 Models and assumptions

Network model: A mesh network is modelled by a *directed* graph $G = (V, E)$, where $V = \{v_1, \dots, v_n\}$ is the set of n nodes and E is the set of possible *directed* communication links. Let $\mathbf{E}^-(u)$ ($\mathbf{E}^+(u)$) denote the set of directed links that end (start) at node u. Every node v_i has a transmission range $R_T(i)$: $\|v_i - v_j\| \le R_T(i)$ is *not* the sufficient condition for $(v_i, v_j) \in E$. Some links do not belong to G because of either the physical barriers or the selection of routing protocols. We always use $\mathbf{L}_{i,j}$ to denote the directed link (v_i, v_j) hereafter. For each link $e = (u, v)$, the maximum rate at which a mesh router u can communicate with the mesh router v in one-hop communication supported by link e is denoted by $\mathbf{c}(e)$. Notice that the links are directed, thus, the capacity could be asymmetric, i.e., $\mathbf{c}((u, v))$ may not be the same as $\mathbf{c}((v, u))$.

Among the set V of all wireless nodes, some of them are gateways which have gateway functionality and provide the connectivity to the Internet. For simplicity, let $\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_k\}$ be the set of k gateway nodes, where $\mathbf{s}_i$ is actually node v_{n+i-k}, for $1 \le i \le k$. All other wireless nodes v_i (for $1 \le i \le n-k$) $\in \overline{\mathcal{S}} = V - S$ are *ordinary* mesh nodes. Each ordinary mesh node u will aggregate the traffic from all its users and then route them to the Internet through some gateway nodes. We assume that the capacity between any gateway nodes to the Internet is sufficiently large. We use $\ell_O(u)$ ($\ell_I(u)$) to denote the total aggregated outgoing (incoming) traffic for its users by mesh node u. We will mainly concentrate on one of the traffic patterns in this paper, i.e., incoming traffic. For notation simplicity, we use $\ell(u)$ to denote such load for node u. Notice that the traffic $\ell(u)$ is not requested to be routed through a specific gateway node, neither requested to be using a single routing path. Our results can be easily extended to deal with both incoming and outgoing traffic by defining routing flows for both traffic patterns separately.

Interference model: Each node v_i also has an interference range $R_I(i)$ such that node v_j is interfered by the signal from v_i whenever $\|v_i - v_j\| \le R_I(i)$ and v_j is not the intended receiver. The interference range $R_I(i)$ is not necessarily same as the transmission range $R_T(i)$. Typically, $R_T(i) < R_I(i) \le c \cdot R_T(i)$ for some constant $c > 1$. We call the ratio between them as the *Interference-Transmission Ratio* for node v_i, denoted as $\gamma_i = \frac{R_I(i)}{R_T(i)}$. In practice, $2 \le \gamma_i \le 4$. For all wireless nodes, let $\gamma = \max_{v_i \in V} \frac{R_I(i)}{R_T(i)}$.

To schedule two links at the same time slot, we must ensure that the schedule will avoid the link interference. Different types of link interference have been studied in the literature, such as *protocol interferences model* (PrIM) [2], *fixed protocol interferences model* (fPrIM) [9, 17], *RTS/CTS model* (RTS-CTS) [8], and *transmitter interference model* (TxIM) [18]. In this paper we adopt the fPrIM by assuming that any node v_j will be interfered by the signal from v_p if $\|v_p - v_j\| \le R_I(p)$ and node v_p is sending signal to some node other than v_j. See Fig. 2a. In other words, the transmission from v_i to v_j is viewed successful if $\|v_p - v_j\| > R_I(p)$ for every node v_p transmitting in the same time slot, as shown in Fig. 2b. Actually, our gateway deployment method can work for any kinds of interference models as we will discuss in Section 6. Given a network $G = (V, E)$, we use the *conflict graph* (e.g., [19]) F_G to



a v_j is interfered by v_p **b** v_j is not interfered by v_p

Figure 2 Illustration of fPrIM interference model

represent the interference in G. Each vertex (denoted by $\mathbf{L}_{i,j}$) of F_G corresponds to a directed link (v_i, v_j) in the communication graph G. There is a *directed edge* from vertex $\mathbf{L}_{i,j}$ to vertex $\mathbf{L}_{p,q}$ in F_G if and only if the transmission of $\mathbf{L}_{i,j}$ interferences the reception of the receiving node of link $\mathbf{L}_{p,q}$. For easy reading, we summarize all used notations in this paper in Appendix (Table 7).

3 Throughput optimization in mesh networks

In this section, we study what is the best throughput achievable by a given multi-hop mesh networks using best possible routing and link scheduling. Here, we assume that the routing between a given mesh router and some gateway nodes can use multiple paths. In practice, we do not need every session to be multi-path. We essentially assume that the aggregated traffic between the mesh router and the gateway nodes could be infinitely divisible. We also assume the time is slotted and synchronized.

Every mesh router u has a traffic demand $\ell(u)$ that needs to be routed to the Internet via some gateway nodes. We want to maximize the total routed traffic to the Internet while certain minimum traffic from each mesh router should be satisfied. Our approach is to give each link $\mathbf{L} \in G$ an interference-aware transmission schedule $\mathcal{S}(\mathbf{L})$ which assigns the time slot for transmission to maximize the overall network throughout. A link scheduling is to assign each link a set of time slots $\subset [1, T]$ in which it can transmit, where T is the scheduling period. A link scheduling is *interference-aware* (or called *valid*) if a scheduled transmission on a link $u \to v$ will not result in a collision at either node u or node v (or any other node) due to the simultaneous transmission of other links. Let $X_{e,t} \in \{0, 1\}$ be the indicator variable which is 1 if and only if e will transmit at time-slot t. We focus on periodic schedules here. A schedule is periodic with period T if, for every link e and time slot t, $X_{e,t} = X_{e,t+i\cdot T}$ for any integer $i \geq 0$. For a link e, let $\mathbf{I}(e)$ denote the set of links e' that will cause interference if e and e' are scheduled at the same time slot. A schedule $\mathcal{S}$ is *interference-free* if $X_{e,t} + X_{e',t} \leq 1$, $\forall e' \in \mathbf{I}(e)$.

We now provide a mixed integer programming formulation of the throughput optimization and a greedy algorithm for interference-free link scheduling. For cross-layer optimization, the flow supported by mesh networks not only needs to satisfy the capacity constraints, but also needs to be schedulable by all links without interference.

3.1 Integer linear programming for throughput optimization

We first formulate the routing problem to maximize the throughput of the achieved flow under certain fairness constraints. Let $\alpha(e) \in [0, 1]$ denote the fraction of the time slots in one scheduling-period that link e is actively transmitting. Obviously, $\alpha(e) \cdot \mathbf{c}(e)$ is the corresponding achieved flow. Given a routing (and corresponding link scheduling), the achieved fairness λ is defined as the minimum ratio of *achieved flow* over the *demanded load* over all wireless mesh routers. Assume that we have a minimum fairness constraint λ_0, then $f(u)$ should satisfy $f(u) \geq \lambda_0 \ell(u)$ for every mesh router u. Clearly, the achieved flow at a router u is the difference between the flow goes out of node u and the flow comes to node u, i.e., $\sum_{e\in\mathbf{E}^+(u)} f(e) - \sum_{e\in\mathbf{E}^-(u)} f(e)$. Here $f(e)$ is the total scheduled traffic over link e. Our goal is to maximize the total throughput which is the summation of traffic flows into all gateways. The maximum throughput routing is equivalent to solve the following linear programming (**LP-Flow-Throughput-1**) for $\alpha(e, \mathbf{f})$ such that

$$\textbf{LP-Flow-Throughput-1:} \quad \max \sum_{i=1}^{k} f(\mathbf{s}_i)$$

$$\begin{cases} \sum_{e\in\mathbf{E}^+(u)} f(e) - \sum_{e\in\mathbf{E}^-(u)} f(e) = f(u) & \forall u \in \overline{\mathcal{S}} \\ f(u) \geq \lambda_0 \ell(u) & \forall u \in \overline{\mathcal{S}} \\ \sum_{e\in\mathbf{E}^-(\mathbf{s}_i)} f(e) - \sum_{e\in\mathbf{E}^+(\mathbf{s}_i)} f(e) = f(\mathbf{s}_i) & \forall \mathbf{s}_i \in \mathcal{S} \\ \alpha(e) \cdot \mathbf{c}(e) = f(e) & \forall e \\ \alpha(e) \geq 0 & \forall e \\ \alpha(e) \leq 1 & \forall e \\ \textit{exists}\text{ interfence-free schedule for } \alpha(e) & \end{cases}$$

Our objective of periodic TDMA link scheduling is to give each link $\mathbf{L} \in G$ a transmission schedule $\mathcal{S}(\mathbf{L})$, which is the list of time-slot that a link can send packets such that the schedule is interference-free. We then mathematically formulate a necessary, sufficient condition for schedulable flow $f(e) = \alpha(e) \cdot \mathbf{c}(e)$: a flow f (equivalently, whether a given vector $\alpha(e)$ for all e is schedulable) is schedulable if and only if we can find integer solution $X_{e,t}$ satisfying the following conditions.

Necessary and Sufficient Condition for Schedulable Flow :

$$\begin{cases} X_{e,t} + X_{e',t} \leq 1 & \forall e' \in \mathbf{I}(e), \forall e, \forall t \\ \frac{\sum_{1 \leq t \leq T} X_{e,t}}{T} = \alpha(e) & \forall e \\ X_{e,t} \in \{0,1\} \ \forall e, \forall t & \end{cases}$$

The first condition says that a schedule should be interference-free. The second condition says that the schedule should achieve the required flow $\alpha(e)$. It is widely known that it is NP-hard to decide whether a feasible scheduling $X_{e,t}$ exists when given the flow $f(e)$ (or equivalently, $\alpha(e)$) for wireless networks with interference constraints. For some interference models several papers gave relaxed necessary conditions and relaxed sufficient conditions for schedulable flows that can be decided in polynomial time. Similar to the proofs in [9, 17], we can prove the following lemma which gives a necessary and a sufficient condition for schedulable flows under fPrIM interference model.

Lemma 1 *Under fPrIM model, consider the active fraction $\alpha(e) \in [0, 1]$ of each link. A sufficient condition that this α is schedulable is, for each e, $\alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq 1$. A necessary condition that this α is schedulable is, for each e, $\alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq C_1$, where $C_1 = \lceil \frac{2\pi}{\arcsin \frac{\gamma - 1}{2\gamma}} \rceil$.*

Here $\mathbf{I}_1(e) \subseteq \mathbf{I}(e)$ denotes the set of links e' that will cause interference at the receiving node of link e if both e and e' are scheduled at the same time slot. For example, in Fig. 7a, $e' = (v_p v_q)$ interferes the receiving node v_j of $e = (v_i v_j)$. Notice that C_1 is a constant for the fPrIM depending on γ, e.g., $C_1 = 25$ when $\gamma = 2$. We provide a detailed proof of this lemma in Appendix. Then we can relax the original mixed integer programming to a linear programming by getting rid of the scheduling variables X. Based on previous studies, given a constant integer $C \in [1, C_1]$, we need to solve the following linear programming (**LP-Flow-Throughput-2**) for $\alpha(e)$ such that

LP-Flow-Throughput-2: $\max \sum_{i=1}^{k} f(\mathbf{s}_i)$

$$\begin{cases} \sum_{e \in \mathbf{E}^+(u)} f(e) - \sum_{e \in \mathbf{E}^-(u)} f(e) = f(u) & \forall u \in \overline{\mathcal{S}} \\ f(u) \geq \lambda_0 \ell(u) & \forall u \in \overline{\mathcal{S}} \\ \sum_{e \in \mathbf{E}^-(\mathbf{s}_i)} f(e) - \sum_{e \in \mathbf{E}^+(\mathbf{s}_i)} f(e) = f(\mathbf{s}_i) & \forall \mathbf{s}_i \in \mathcal{S} \\ \alpha(e) \cdot \mathbf{c}(e) = f(e) & \forall e \\ \alpha(e) \geq 0 & \forall e \\ \alpha(e) \leq 1 & \forall e \\ \alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq C & \forall e \end{cases}$$

3.2 Interference-free link scheduling

Interference-aware link scheduling for wireless networks has been studied in [17]. Here, we apply a classical greed method to design efficient link scheduling that can achieve $\alpha(e)$ found from the solution of the LP. Assume that we already have the values $\alpha(e)$ for every links e and T is the number of time slots per scheduling period. Then we need to schedule $T \cdot \alpha(e)$ time-slots for a link e. For simplicity, we assume that the choice of T results that $T \cdot \alpha(e)$ is an integer for every e. Notice that when we schedule each link, we need to ensure that the scheduling is interference-free. Algorithm 1 illustrates our scheduling method. The basic idea of our scheduling is first sorting the links based on some specific order and then process the requirement $\alpha(e)$ for each link in a greedy manner. When process the i^{th} link e_i, we assign link e_i the *earliest* (no need to be consecutive) $N(e_i) = T \cdot \alpha(e_i)$ time slots that will not cause any interference to already scheduled links.

Algorithm 1 Greedy link scheduling

Input: A communication graph $G = (V, E)$ of m links and $\alpha(e)$ for all links.

Output: An interference-free link scheduling.

Sort the links in the communication graph G using the following method:
Consider the conflict graph F_G. We choose the vertex, which is the link in the original graph, with the largest value $d^{\text{in}}_{i,j} - d^{\text{out}}_{i,j}$ in the residue conflict graph; remove the vertex and its incident edges. Here, $d^{\text{in}}_{i,j}$ and $d^{\text{out}}_{i,j}$ are the *in-degree* and *out-degree* of vertex $\mathbf{L}_{i,j}$ in the conflict graph under fPrIM model. Repeat this process until there is no vertex in the conflict graph. Then the links (in the original graph) are sorted by their reverse removal order. Let $(e_1, e_2, \cdots, e_m)$ be the sorted list of links.
Assign the time slot using the following greedy method:
for $i = 1$ to m **do**
  $N(e_i) = T \cdot \alpha(e_i)$ be the number of time slots that link e_i will be active.
  Assume $e_i = (u, v)$. Set *allocated* $\leftarrow 0$; $t \leftarrow 1$;
  while *allocated* $< N(e_i)$ **do**
    if $X_{e',t} = 0$ for every conflicting link $e' \in \mathbf{I}_1(e_i)$, $\sum_{e':e' \ni u} X_{e',t} < 1$, $\sum_{e':e' \ni v} X_{e',t} < 1$ **then**
      Set $X_{e_i,t} \leftarrow 1$; Set *allocated* $\leftarrow$ *allocated* $+ 1$;
    end if
    Set $t \leftarrow t + 1$.
  end while
end for

We can prove the following theorems regarding Algorithm 1.

Theorem 2 *Algorithm* 1 *produces a feasible interference-free link-channel scheduling when* $\alpha(e)$ *is a feasible solution of LP using* $C = 1$.

Proof Assume that from the Linear Program **LP-Flow-Throughput-2**, we get the solution $\alpha(e)$. Essentially we need to show that Algorithm 1 will terminate. Notice that after the algorithm terminates, we know that for every link e, it has already been assigned a fraction $\alpha(e)$ time slot in a schedule period T. Consider a specific link e that is to be processed. Based on the special sorting used by our algorithm (generated by Step 1-2) for fPrIM, we know that all links e' that have been processed and conflict with e (interfering e or being interfered by link e) must be a subset of $\mathbf{I}_1(e)$. Recall that in our linear programming, we had a condition that, $\alpha(e) + \sum_{(e')\in\mathbf{I}_1(e)} \alpha(e') \leq 1$. This implies that,

$$N(e) + \sum_{(e')\in\mathbf{I}_1(e)} N(e') \leq T,\ \forall e.$$

Thus, we can always find $N(e) = T \cdot \alpha(e)$ time-slots among T slots in a period for link e, since all conflict links that have already been processed by Algorithm 1 occupy at most $\sum_{(e')\in\mathbf{I}_1(e)} N(e') \leq T - N(e)$ time slots. Because the total number of time slots needed for a node u_i is $\sum_{e:u_i\in e} T \cdot \alpha(e) \leq T$, among T time slots, we can always find time slots for link e (after considering all conflicting links scheduled before). This finishes our proof. □

Theorem 3 *Algorithm* 1*, together with the linear programming formulation* ***LP-Flow-Throughput-2****, produces a feasible interference-free link-channel scheduling whose achieved throughput is at least* $\frac{1}{C_1}$ *of the optimum, and the fairness is at least* $\frac{\lambda_0}{C_1}$ *(instead of the required* λ_0*), when* $\alpha(e)$ *is a feasible solution of LP using* $C = 1$.

Proof Consider an optimum flow assignment defined by $\alpha^*(e)$, i.e., the flow supported by a link e is $\alpha^*(e) \cdot \mathbf{c}(e)$. From Lemma 1, we know that $\alpha^*(e) + \sum_{(e')\in\mathbf{I}_1(e)} \alpha^*(e') \leq C_1$. Define a new flow α' as $\alpha'(e) = \frac{\alpha^*(e)}{C_1}$. Obviously, $\alpha'(e) + \sum_{(e')\in\mathbf{I}_1(e)} \alpha'(e') \leq 1$. It is easy to show that the new flow α' satisfies all conditions of our linear programming **LP-Flow-Throughput-2**. In other words, α' is a feasible solution for this LP. Consequently, the solution of **LP-Flow-Throughput-2** is at least that of α', which is $\frac{1}{C_1}$ of the optimum. □

4 Gateways placement schemes

We provide a method (Algorithm 1 together with the linear programming formulation **LP-Flow-Throughput-2**) in Section 3 to achieve an interference-free link scheduling which maximizes the network throughput. In other words, this method can be used to evaluate a fixed mesh networks with certain gateways in term of throughput optimization. In this section, we propose a grid-based gateway placement scheme which uses the linear programming **LP-Flow-Throughput-2** as a evaluation tool. The problem we want to study is as follows:

The Problem: Given a mesh network with $n - k$ fixed mesh nodes and interference model, our gateway placement needs to select positions for k gateways in order to maximize the throughput. It is clear that we can not try all possible positions since the possible combination is infinite.

4.1 Three gateways placement schemes

Random deployment: The easiest and simplest method is random deployment where we randomly select k positions for gateways. For example, Fig. 3a shows four gateways are deployed randomly in a mesh network. However, the random deployment maybe not good at

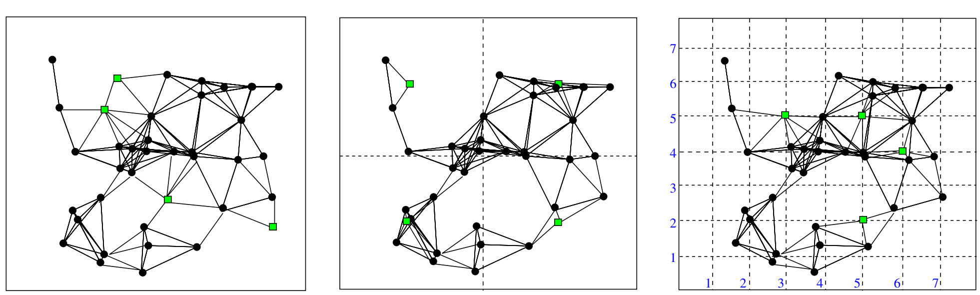


Figure 3 Three gateway deployment methods: four gateways (*grey square*) are deployed in a mesh network with 33 mesh nodes (*black dot*)

a Random Deployment **b** Fixed Deployment **c** Grid-based Deployment the throughput or even can not guarantee the connectivity of the mesh network.

Fixed deployment: The second method is to deploy the gateways in fixed positions which are the centers of evenly distributed cells. As shown in Fig. 3b, to place four gateways, we divide the whole area into four cells and put the gateways in the centers of these cells. This fixed deployment scheme should be able to work well with well-spread and evenly-distributed mesh networks. However, if the network is not so even, for example, putting a gateway at the center of the upper-left cell in Fig. 3b does not help a lot for the throughput since the gateway can only connect 2 mesh nodes and one of them is an end-point. In the real-life applications, the mesh network usually is not evenly-distributed. For example, houses are arbitrarily distributed in a neighborhood due to different designs and various landscapes (e.g., a lake or a hill).

Grid-based deployment: To explore more choices of gateway layouts but at the same time to keep the scalability of the method, we propose a new grid-based deployment scheme. The idea is simple. The whole deployment area is divided into an $a \times b$ grid. As shown in Fig. 3c, which is a 7×7 grid, we only place the gateways in the cross points on this grid. We will try all possible combinations of the k-gateway placement, and evaluate each of them using the method in previous section (computing the maximum throughput can be achieved by this combination). Finally, we select the placement which has the largest maximum throughput. For an $a \times b$ grid, the number of total combinations is $C^k_{a\times b}$ which is the combination of selecting k elements from $a \times b$ elements. Even though this number could be large, it is still reasonable to try all of them since the deployment scheme will only run once before the real gateway installation and the positions of all mesh routers are fixed. In addition, the overhead cost depends on the size of the grid. It is an adjustable parameter which can be easily controlled for the tradeoff between computation overhead and throughput performance. If both a and b goes to infinite, our grid-based method can potentially explore all possible deployment layouts.

We will test all these three methods by conducing simulations with random networks in Section 5.

4.2 Performance guarantee

In this section, we will provide performance analysis of our grid based deployment. For simplification, we assume that the deployment area is a $l \times l$ square, and our grid-based method use a $a \times a$ grid. Thus the length of each cell in the grid is $\frac{l}{a+1}$. We use $\Omega(S)$ to denote the total throughput achieved by the solution S (which gives the positions of all gateways). We assume that the total throughput of mesh gateways is *Lipschitz continuity* within the deployment area. Lipschitz continuity is a smoothness condition for functions. A function $g()$ is Lipschitz with a coefficient β if for any two points x and y in the domain $|g(x) - g(y)| \le \beta||x - y||$. Here, we assume that the throughput of mesh gateways $\Omega(S)$ is Lipschitz with a coefficient β. In other words, given a set of position of k gateways ($S = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_k\}$), if we move the position of $\mathbf{s}_i$ to $\mathbf{s}'_i$, then the change of the achievable total maximum throughput is bounded by $\beta||\mathbf{s}_i\mathbf{s}'_i||$ where $||\mathbf{s}_i\mathbf{s}'_i||$ is the distance between $\mathbf{s}_i$ and $\mathbf{s}'_i$. We call this assumption *Lipschitz-throughput assumption*. We did not verify whether this assumption is valid for real mesh networks, but we believe the assumption is reasonable for the simplification of theoretical analysis. Then we can prove the following theorem for our grid-based method.

Theorem 4 *Under the* Lipschitz-throughput assumption*, our grid-based deployment can achieve the total throughput at least the optimal minus $k \cdot \beta \cdot b$, where k is the number of gateways, β is Lipschitz coefficient, and b is a constant depending on the size of the grid.*

Proof Assume that the optimal solution $OPT = \mathbf{s}_1^{OPT}, \mathbf{s}_2^{OPT}, \cdots, \mathbf{s}_k^{OPT}$ where $\mathbf{s}_i^{OPT}$ is the ith gateway in the solution and the throughput achieved by OPT is $\Omega(OPT) = \sum_{i=1}^k f(\mathbf{s}_i^{OPT})$. For each $\mathbf{s}_i^{OPT}$, we can define a grid node $\mathbf{s}_i^{GEO}$ which is nearest to $\mathbf{s}_i^{OPT}$. Then we denote the union of all such grid nodes by GEO (grid estimation of OPT), which is also a solution of positions for k gateways. See Fig. 4 for illustration. Notice that the distance between $\mathbf{s}_i^{OPT}$ and $\mathbf{s}_i^{GEO}$ must be smaller than $b = \frac{\sqrt{2}}{2} \cdot \frac{l}{a+1}$. Due to the Lipschitz continuity of throughput, we have

$$\Omega(OPT) - \Omega(GEO) \le k \cdot \beta \cdot b.$$

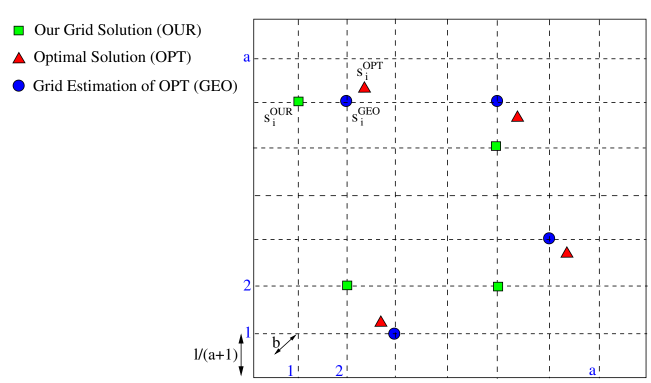


Figure 4 Illustration for the proof of Theorem 4

Thus, $\Omega(GEO) \geq \Omega(OPT) - k \cdot \beta \cdot b$. Notice that to get OPT we need move k gateways in GEO from $\mathbf{s}_i^{GEO}$ to $\mathbf{s}_i^{OPT}$. If we move the gateway one by one, each time the change of total throughout is bounded by $\beta \cdot b$ since the distance of each move is bounded by b.

Assume that our grid-based deployment generate a set of solution $OUR = \mathbf{s}_1^{OUR}, \mathbf{s}_2^{OUR}, \cdots, \mathbf{s}_k^{OUR}$. Since we take the maximum throughput among all combinations of grid positions, we have

$$\Omega(OUR) \geq \Omega(GEO).$$

Consequently, we have

$$\Omega(OUR) \geq \Omega(GEO) \geq \Omega(OPT) - k \cdot \beta \cdot b.$$

This finishes the proof. □

Theorem 4 shows that our solution can achieve almost the same throughput as the optimal one if the size of the cell is very small. After we select the gateways positions using the grid-based method, we can use Algorithm 1 to schedule the traffic. From Theorem 3, we know the total throughput achieved by our method is $\frac{1}{C_1}$ of the optimal of given fixed gateways. Putting together with results from Theorem 4, our method can achieve throughput $\frac{1}{C_1}(\Omega(OPT) - k \cdot \beta \cdot b)$. Remember C_1 is a constant depending on the interference model, k is the number of gateways, and b is a constant depending on the size of the grid.

5 Simulations

In this section, we evaluate the maximal flow of different gateway deployment schemes in random wireless mesh networks. As we have discussed in Section 3, the maximal flow is solved by a linear programming. The

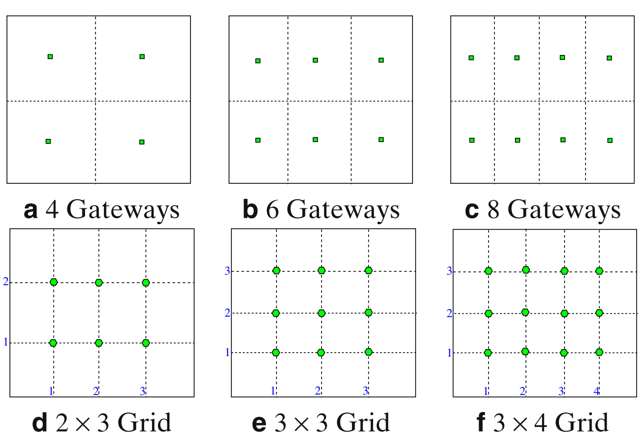


Figure 5 The layouts of gateways in fixed deployment scheme (**a**–**c**) and the grids used in OUR deployment scheme (**d**–**f**)

Table 1 Avg throughput (various network sizes) when $\lambda_0 = 0.2$

Nodes	Gateways	Random	Fixed	3 × 4 Grid
60	6	551.3	677.6	831.2
80	6	686.1	845.5	959.9
100	6	783.2	947.5	1,082.0

wireless mesh network in our simulation is randomly generated, i.e., the positions of n mesh nodes are randomly chosen in certain area. For each generated mesh network, the deployment method will decide how to place k gateways to connect the mesh routers to the Internet. We use 802.11a for the link channel capacity in the wireless mesh network, which is the same as [8]. The link channel capacity thus only depends on the distance between the two nodes at the end of each link. We set the link channel capacity as 54 Mbps when the distance of the two end nodes is within 30 m, 48 Mbps when the distance is within 32 m, 36 Mbps when the distance is within 37 m, 24 Mbps when the distance is within 45 m, 18 Mbps when the distance is within 60 m, 12 Mbps when the distance is within 69 m, 9 Mbps when the distance is within 77 m, and 6 Mbps when the distance is within 90 m. Otherwise, if the distance of the two end nodes of the link is beyond 90 m, we will set the link channel capacity as 0. Each node has 180 m interference range. The wireless mesh network is generated with 60–100 mesh routers and four to eight gateways. The mesh routers are randomly dispersed in a square area of 500 × 500 m^2. Each mesh router transfers 20 Mbps data to the Internet. The input value of λ_0 and C in the LP to solve the maximal throughput is set as 0.2 and 20.

We evaluate three gateway deployment schemes described in Section 4. The fixed deployment scheme first divides the square area into k equal cells as shown in Fig. 5a–c, and then put the k gateways in the centers of these cells. Our grid-based deployment scheme will use various grids defined in Fig. 5d–f to define the candidate positions of gateways, and then try all the combinations of positions using the LP to evaluate their throughput, and select the combination with highest throughput.

Table 2 Avg throughput (various numbers of gateways) when $\lambda_0 = 0.2$

Nodes	Gateways	Random	Fixed	3 × 4 Grid
60	4	393.4	442.6	648.7
60	6	551.3	677.6	831.2
60	8	721.1	854.4	952.3

Table 3 Avg throughput (various grid sizes) when $\lambda_0 = 0.2$

Nodes	Gateways	2×3 Grid	3×3 Grid	3×4 Grid
60	6	681.6	756.3	820.5
80	6	787.6	852.8	944.3
100	6	962.5	977.3	1083.8

Table 5 Avg throughput without fairness (various numbers of gateways) when $\lambda_0 = 0$

Nodes	Gateways	Random	Fixed	3×4 Grid
60	4	406.4	421.9	642.8
60	6	561.3	659.1	823.9
60	8	729.7	829.1	952.2

We vary the numbers of mesh routers, mesh gateways and cells of the grid to test the performance of these three deployment schemes. Each data in Tables 1, 2 and 3 is the average number computed over all 100 random networks.

Table 1 shows the results for networks with 60, 80 and 100 mesh routers and 6 gateways to be deployed. It is clear that our grid-based method can achieve better throughput than the random and fixed schemes. Notice that there are many cases that the random deployment method can not find feasible solutions in LP or even can not form a connected mesh network. We exclude those cases in the results presented in Table 1 and 2. In other words, all the data here are for the mesh network where the random deployment can find the feasible solution.

Table 2 shows the results when we want to deploy various number of gateways. The number of gateways is from 4 to 8 when the number of mesh routers are fixed at 60. It is clear that with more gateways the performance is better.

Table 3 shows the results when we increasing the size of the grid from 2×3 to 3×4 when the number of gateway is fixed at 6. Here, we do not request the network needs to have feasible solution for the random deployment. Thus, the data in Table 3 are different from the data in Tables 1 and 2, even though the number of gateways, nodes and grids are the same. It is clear that the larger size of grid can improve the throughput, but also increases the computation cost. Therefore, in practice, the administrator needs to find an appropriate grid to satisfy both performance and cost requirements. On the other hand, by having the ability to change the grid size, it gives the way for administrator to play with the tradeoff.

Notice that there are many cases that the certain deployment method (especially for random deployment) can not find feasible solutions in LP due to the following fairness constraint:

$$f(u) \geq \lambda_0 \ell(u), \forall u \in \overline{\mathcal{S}}.$$

Thus, we also perform a new set of simulations by removing the fairness constraint, i.e. set $\lambda_0 = 0$. This can guarantee that we have solutions in LP. Again, we vary the numbers of mesh routers, mesh gateways and cells of the grid to test the performance of all three deployment schemes. Each data in Tables 4, 5 and 6 is the average number computed over all 100 random networks. The out-performance of our grid-based method is also very clear.

6 Discussions

So far, we only consider the network with a single channel and using fPrIM model. However, our gateway placement method based on throughput optimization can be extended for various networks with different models.

Various interference models: Our maximum throughput method can be extended to deal with different interference models, such as PrIM [2], RTS-CTS [8], and TxIM [18]. The differences of these models with the fPrIM are that they have different definitions of link interference. The only changes needed in our method are (1) the sorting method in Step 1–2 of Algorithm 1, and (2) the constant C_1. In [9], the authors showed how to do the sorting under different interference models for link scheduling and provided the values of C_1 for those models.

Multi-channel and multi-radio networks: A number of schemes [20–23] have been proposed recently to

Table 4 Avg throughput without fairness (various network sizes) when $\lambda_0 = 0$

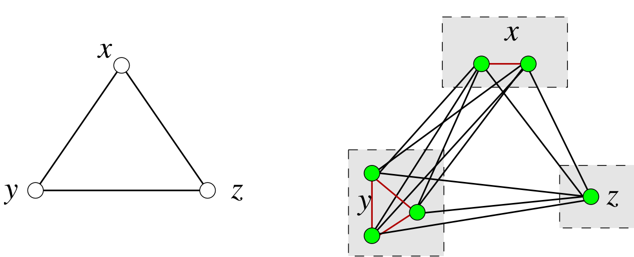
Nodes	Gateways	Random	Fixed	3×4 Grid
60	6	561.3	659.1	823.9
80	6	717.2	836.4	991.2
100	6	855.9	1013.4	1133.2

Table 6 Avg throughput without fairness (various grid sizes) when $\lambda_0 = 0$

Nodes	Gateways	2×3 Grid	3×3 Grid	3×4 Grid
60	6	652.0	782.0	842.6
80	6	855.3	921.4	1011.0
100	6	1003.9	1032.6	1131.7

exploit multiple channels and multiple radios for performance improvement in wireless mesh networks. Using multiple channels and multiple radios can alleviate but not eliminate the interference. For multi-channel and multi-radio mesh networks, we can first convert the network model (the graph model) G to a single-radio and multi-channel graph model G', then refine our linear programming for throughput optimization by define the fraction of flow for each pair of link e and channel $\mathbf{f}$ instead of just e. Notice that a similar idea has been proposed in [25] for joint routing and channel assignment in multi-radio mesh networks.

The method of converting works as follows. Let $\mathcal{F}$ be the set of orthogonal channels that can be used by all wireless nodes. Each wireless node u is equipped with $\mathcal{I}(u) \geq 1$ radio interfaces. Each wireless node u can only operate on a subset of channels $\mathcal{F}(u)$ from $\mathcal{F}$ due to the hardware constraints. For each node u, we split it into $\mathcal{I}(u)$ pseudo nodes $u_1, u_2, \cdots, u_{\mathcal{I}(u)}$ in G'. For notational convenience, we use $\mathcal{F}(e)$ to denote the set of common channels among $\mathcal{F}(u)$ and $\mathcal{F}(v)$ for any link $e = (u, v)$ in G. For $e = (u, v)$ in G, we connect u_i and v_j using $e' = (u_i, v_j)$ in G' if ith interface of u and jth interface of v share some common channels denoted by $\mathcal{F}(e')$. We also interconnect all pseudo nodes u_i of u to each other using links with infinite capacity. See Fig. 6 for illustration of an example in which $\mathcal{I}(x) = 2$, $\mathcal{I}(y) = 3$, and $\mathcal{I}(z) = 1$. Then we let $\delta(e, \mathbf{f}) \in \{0, 1\}$ be the indicator function whether a channel $\mathbf{f}$ can be used by a link e in G'. For each link $e = (u_i, v_j)$ operating on a channel $\mathbf{f} \in \mathcal{F}(e)$, we denote by $\mathbf{c}(e, \mathbf{f})$ the rate for link e in G'. This is the maximum rate at which a mesh router u's ith interface can communicate with the mesh router v's jth interface in one-hop communication using channel $\mathbf{f}$. Let $\alpha(e, \mathbf{f}) \in [0, 1]$ denote the fraction of the time slots in one scheduling-period that link e is actively transmitting using channel $\mathbf{f}$. Obviously, $\alpha(e, \mathbf{f}) \cdot \mathbf{c}(e, \mathbf{f})$ is the corresponding achieved flow.



a original graph G **b** new graph G'

Figure 6 By splitting node with multi-radio interfaces into pseudo nodes, we convert the original communication graph G to a new graph G' without multi-radio. Here, $\mathcal{I}(x) = 2, \mathcal{I}(y) = 3$, and $\mathcal{I}(z) = 1$. The *pseudo nodes in one shaded region* correspond to a node in the original network

We now can refine our linear programming for throughput optimization. The conditions for each node u (including s_i) are still the same, but now we have them for each pseudo node u_i. For each pair of link $e \in G'$ and channel $\mathbf{f}$, we then have $\sum_{\mathbf{f} \in \mathcal{F}(e)} \alpha(e, \mathbf{f}) \cdot \mathbf{c}(e, \mathbf{f}) = f(e)$ and $0 \leq \alpha(e, \mathbf{f}) \leq \delta(e, \mathbf{f}) \leq 1$. In addition, due to interference, $\alpha(e, \mathbf{f}) + \sum_{e', \mathbf{f}' \in \mathbf{I}_1(e, \mathbf{f})} \alpha(e', \mathbf{f}) \leq C$ for each pair of e and $\mathbf{f}$. Here $\mathbf{I}_1(e, \mathbf{f})$ is the set of pairs of link e' and channel $\mathbf{f}'$ that interfere with the link e on channel $\mathbf{f}$, which includes both the links e' operate on the same channel $\mathbf{f}$ or the links e' which are the same link as e in the original G and operate on different $\mathbf{f}'$. Therefore, given a constant integer $C \in [1, C_1]$, we need to solve the following linear programming (**LP-Flow-Throughput-3**) for $\alpha(e, \mathbf{f})$ such that

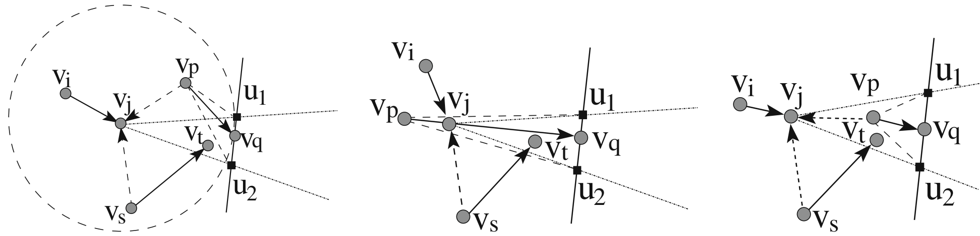
$$\textbf{LP-Flow-Throughput-3:} \quad \max \sum_{i=1}^{k} f(\mathbf{s}_i)$$

$$\begin{cases} \sum_{e \in \mathbf{E}^+(u)} f(e) - \sum_{e \in \mathbf{E}^-(u)} f(e) = f(u) & \forall u \in \overline{\mathcal{S}} \\ f(u) \geq \lambda_0 \ell(u) & \forall u \in \overline{\mathcal{S}} \\ \sum_{e \in \mathbf{E}^-(\mathbf{s}_i)} f(e) - \sum_{e \in \mathbf{E}^+(\mathbf{s}_i)} f(e) = f(\mathbf{s}_i) & \forall \mathbf{s}_i \in \mathcal{S} \\ \sum_{\mathbf{f} \in \mathcal{F}(e)} \alpha(e, \mathbf{f}) \cdot \mathbf{c}(e, \mathbf{f}) = f(e) & \forall e \\ \alpha(e, \mathbf{f}) \geq 0 & \forall e \\ \alpha(e, \mathbf{f}) \leq 1 & \forall e \\ \alpha(e, \mathbf{f}) + \sum_{e' \in \mathbf{I}_1(e, \mathbf{f})} \alpha(e', \mathbf{f}) \leq C & \forall e \\ \alpha(e, \mathbf{f}) \leq \delta(e, \mathbf{f}) & \\ \sum_{e \ni u, \mathbf{f}} \alpha(e, \mathbf{f}) \leq \mathcal{I}(u) & \end{cases}$$

Algorithm 2 Greedy link scheduling for multi-radio multi-channel networks

Input: The converted communication graph $G' = (V, E)$ of m links and $\alpha(e, \mathbf{f})$ for all links and channels.
Output: An interference-free link scheduling.

```
Sort the links in G as the same in Algorithm 1. Let
  (e_1, e_2, ..., e_m) be the sorted list of links.
for i = 1 to m do
  for each possible channel f ∈ F do
    Let N(e_i, f) = T · α(e_i, f) be the number of
      time slots that link e_i will be active using channel f.
    Assume e_i = (u, v). Set allocated ← 0; t ← 1;
    while allocated < N(e, f) do
      if X_{e',t,f} = 0 for every conflicting link e' ∈ I_1(e_i),
        Σ_{f,e':e'∋u} X_{e',t,f} < I(u), Σ_{f,e':e'∋v} X_{e',t,f} < I(v)
      then
        Set X_{e_i,t,f} ← 1; Set allocated ← allocated + 1;
      end if
      Set t ← t + 1.
    end while
  end for
end for
```



a Case 1 **b** Case 2 **c** Case 3

Figure 7 Links in a small neighborhood will interfere with each other in fixed protocol interference model (fPrIM)

The link scheduling in multi-channel and multi-radio mesh networks also needs to satisfy the channel and radio constraints no matter whether dynamic channel assignment or fixed channel assignment is used. The greedy link scheduling (Algorithm 1) can also be extended to schedule the links and channels. The detailed algorithm is given by Algorithm 2. The basic idea is as follows. When processing the ith link $e_i \in G'$, we process the channels in order and assign link e_i the earliest $N(e_i, \mathbf{f}) = T \cdot \alpha(e_i, \mathbf{f})$ time slots using channel $\mathbf{f}$ that will not cause any interference to already scheduled links, and satisfy the radio and channel-availability constraints.

By combining the Algorithm 2 with the linear programming formulation **LP-Flow-Throughput-3**, we can produce an interference-free link-channel scheduling for multi-channel and multi-radio mesh networks. It is easy to extend the proofs of Theorem 2 and Theorem 3 to the multi-channel and multi-radio case. In other words, we can generate a feasible interference-free link-channel scheduling whose achieved throughput is at least $\frac{1}{C_1}$ of the optimum, and the fairness is at least

Table 7 Notation used

Term	Definition
V, E	Set of n nodes ($V = \{v_1, \ldots, v_n\}$) and set of possible *directed* communication links
$\mathcal{S}$, $\overline{\mathcal{S}}$	Set of k gateway nodes ($\mathcal{S} = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_k\}$) and set of *ordinary* mesh nodes ($\overline{\mathcal{S}} = V - S$)
G	*Directed* communication graph $G = (V, E)$
F_G	Conflict graph, represent the interference in G
$\mathbf{E}^-(u)$, $\mathbf{E}^+(u)$	Set of directed links that end (start) at node u
$R_T(i)$, $R_I(i)$	Transmission range and interference range of node v_i
γ	Interference-transmission ratio, $\gamma = \max_{v_i \in V} \frac{R_I(i)}{R_T(i)}$, $\gamma_i = \frac{R_I(i)}{R_T(i)}$ for node v_i
$\mathbf{L}_{i,j}$	Directed link (v_i, v_j) in G; also vertex in F_G corresponds to directed link (v_i, v_j) in G
$\mathbf{c}(e)$	Capacity of link e
$\ell(u)$	Demanded traffic load for node u
T	Scheduling period
$X_{e,t}$	Indicator variable whether e transmits at time slot t
$\mathbf{I}(e)$	Set of links e' that interfere with edge e
$\mathbf{I}_1(e)$	Set of links e' that interfere with e at the receiving node of e
$\alpha(e)$	Fraction of time slots in one scheduling-period T that link e is active
$f(e)$	Total scheduled traffic over link e, $f(e) = \alpha(e) \cdot \mathbf{c}(e)$
λ	Achieved fairness, i.e., minimum ratio of achieved flow over demanded load among all mesh routers
λ_0	Minimum fairness constraint
C_1	Constant for fPrIM interference model, $C_1 = \lceil \frac{2\pi}{\arcsin \frac{\gamma-1}{2\gamma}} \rceil$
$N(e)$	Number of time slots that link e is active, $N(e) = T \cdot \alpha(e)$
$\mathcal{I}(u)$	Number of radio interfaces node u equipped
$\mathcal{F}(u)$, $\mathcal{F}(e)$	Set of channels node u can use, set of common channels among two endpoints of link e
G'	Single-radio graph model converted from multi-radio graph model G
$\mathbf{c}(e, \mathbf{f})$, $\alpha(e, \mathbf{f})$, $\mathbf{I}_1(e, \mathbf{f})$, $N(e_i, \mathbf{f})$	Capacity, fraction of active slots, set of interfering links, and number of active slots for link e with channel $\mathbf{f}$
$X_{e,t,\mathbf{f}}$	Indicator variable whether e transmits on channel $\mathbf{f}$ at time slot t
$\delta(e, \mathbf{f})$	Indicator function whether a channel $\mathbf{f}$ can be used by a link e in G'

$\frac{\lambda_0}{C_1}$, when $\alpha(e)$ is a feasible solution of LP using $C = 1$ for multi-channel and multi-radio mesh networks.

7 Conclusion

The positions of gateways in wireless mesh networks affect the total network throughput. In this paper, we studied how to place k gateways for a mesh network so that the total throughput achieved by interference-free scheduling is maximized. We proposed a novel grid-based gateway deployment method using a cross-layer throughput optimization and prove that the achieved throughput by our method is a constant times of the optimal. Our simulation results demonstrated that our method achieves better throughput than both random deployment and fixed deployment methods. Furthermore, our proposed method can be extended to work with multi-channel and multi-radio networks under various interference models.

Acknowledgements The authors would like to thank the reviewers for pointing out references [24] and [25]. The work of Yu Wang was supported in part by the US National Science Foundation (NSF) under Grant No. CNS-0721666. The work of Xiang-Yang Li was supported in part by the US National Science Foundation (NSF) under Grant No. CCR-0311174.

Appendix

Lemma 1 *Under fPrIM model, consider the active fraction* $\alpha(e) \in [0, 1]$ *of each link. A sufficient condition that this* α *is schedulable is, for each* e, $\alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq 1$. *A necessary condition that this* α *is schedulable is, for each* e, $\alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq C_1$, *where* $C_1 = \lceil \frac{2\pi}{\arcsin \frac{\gamma-1}{2\gamma}} \rceil$.

Proof The sufficient condition comes directly from the correctness of Algorithm 1 which gives a valid link-channel schedule. Thus, we will only concentrate on the correctness of the necessary condition.

To prove that any valid interference-free link scheduling $\mathcal{S}$ under fPrIM must satisfy that $\alpha(e) + \sum_{e' \in \mathbf{I}_1(e)} \alpha(e') \leq C_1$ for each e, we only need to prove that for all incoming neighboring links of link e there are at most $\lceil \frac{2\pi}{\arcsin \frac{\gamma-1}{2\gamma}} \rceil$ links that can be scheduled at any same time slot. Recall that here $\mathbf{I}_1(e)$ is the set of incoming links of e that interfere e.

Consider any communication link $\mathbf{L}_{i,j}$, where v_j is the receiver. Consider two links $\mathbf{L}_{p,q}$ and $\mathbf{L}_{s,t}$ that are $\mathbf{L}_{i,j}$'s incoming links in conflict graph F_G, where v_q and v_t are the receivers. We now prove that if $\angle v_q v_j v_t \leq \arcsin \frac{\gamma-1}{2\gamma}$, then link $\mathbf{L}_{p,q}$ interferes with link $\mathbf{L}_{s,t}$. This will complete the proof of this lemma.

Draw two rays $v_j v_a$, $v_j v_b$ emanated from node v_j such that $\angle v_a v_j v_b = \arcsin \frac{\gamma-1}{2\gamma}$ and v_q, v_t are in the cone as shown in Fig. 7a. Without loss of generality, we assume that $\|v_j - v_q\| \geq \|v_j - v_t\|$. Draw a circle $\mathcal{C}$ centered at v_j with radius $\|v_j - v_q\|$. Let u_1u_2 be the line passing v_q that is tangent to circle $\mathcal{C}$ and u_1, u_2 are the intersections of this line with line $v_j v_a$, $v_j v_b$ respectively. Since $\angle u_1 v_j v_q \leq \arcsin \frac{\gamma-1}{2\gamma}$, we have

$$\|u_1 - v_q\| \leq \|v_j - v_q\| \cdot \frac{\gamma-1}{2\gamma} \leq 2r_p \cdot \frac{\gamma-1}{2\gamma} = r_p \cdot \frac{\gamma-1}{\gamma}.$$

Thus, $\|v_p - u_1\| \leq \|v_p - v_q\| + \|u_1 - v_q\| \leq r_p \cdot \frac{1}{\gamma} + r_p \cdot \frac{\gamma-1}{\gamma} = r_p$. Similarly, $\|v_p - u_2\| \leq r_p$. Following we prove that node v_p interferes with v_t by cases.

Case 1 $v_p u_1 u_2 v_j$ is a convex quadrangle as shown in Fig. 7a. In this case, v_t is either inside triangle $v_p v_j u_2$ or triangle $v_p u_1 u_2$. Since both $\|v_p - u_1\|$, $\|v_p - u_2\|$ and $\|v_p - v_j\|$ are not greater than r_p, we have $\|v_p - v_t\| \leq r_p$.

Case 2 v_j is inside $\triangle u_1 u_2 v_p$ as shown in Fig. 7b. In this case, v_t is inside triangle $\triangle u_1 u_2 v_p$. Then it is easy to show that $\|v_p - v_t\| \leq \max\{\|v_p - u_1\|, \|v_p - u_2\|\} \leq r_p$.

Case 3 v_p is inside $\triangle u_1 u_2 v_j$ as shown in Fig. 7c. In this case, v_t is inside one of the three triangles: $\triangle u_1 u_2 v_p$, $\triangle u_1 v_j v_p$, $\triangle u_2 v_j v_p$. Similarly, we have $\|v_p - v_t\| \leq r_p$.

Obviously, the above three cases covers all possible situations. This proves that link $\mathbf{L}_{p,q}$ interferes with $\mathbf{L}_{s,t}$. □

For easy reading, we summarize all used notations in this paper in Table 7.

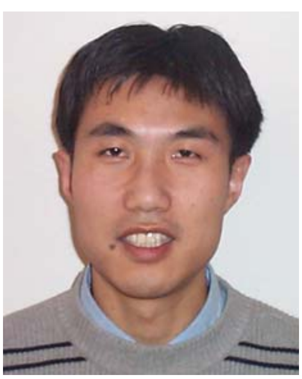
References

1. Ian XW, Akyildiz F, Wang W (2005) Wireless mesh networks: a survey. Comput Netw 47(4):445–487
2. Gupta P, Kumar P (2000) Capacity of wireless networks. IEEE Trans Inf Theory 46(2):388–404, March
3. Li J, Blake C, Couto DSD, Lee HI, Morris R (2001) Capacity of ad hoc wireless networks. In: MobiCom '01: proceedings of the 7th annual international conference on Mobile computing and networking. ACM Press, New York, NY, USA, pp 61–69

4. Grossglauser M, Tse D (2001) Mobility increases the capacity of ad-hoc wireless networks. In: Proc. of IEEE INFOCOMM, vol 3, pp 1360–1369
5. Kyasanur P, Vaidya NH (2005) Capacity of multi-channel wireless networks: impact of number of channels and interfaces. In: MobiCom '05: proceedings of the 11th annual international conference on mobile computing and networking. ACM Press, New York, NY, USA, pp 43–57
6. Kodialam M, Nandagopal T (2003) Characterizing achievable rates in multi-hop wireless networks: the joint routing and scheduling problem. In: MobiCom '03: proceedings of the 9th annual international conference on mobile computing and networking. New York, NY, USA, ACM Press, pp 42–54
7. Kodialam M, Nandagopal T (2005) 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. Cologne, Germany, ACM Press, pp 73–87
8. Alicherry M, Bhatia R, Li LE (2005) Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In: MobiCom '05: proceedings of the 11th annual international conference on mobile computing and networking. ACM Press, New York, NY, USA, pp 58–72
9. Li X-Y, Wu Y, Wang W (2007) Joint throughput optimization for next generation wireless mesh networks. (submitted for publication)
10. Maksuriwong K, Varavithya V, Chaiyaratana N (2003) Wireless LAN access point placement using multi-objective genetic algorithm. In: Proceedings of IEEE international conference on systems, man & cybernetics
11. Kouhbor S, Ugon J, Rubinov A, Kruger A, Mammadov M (2006) WLAN coverage planning: optimization models and algorithms. In: Proceedings of the first IEEE international conference on wireless broadband and ultra wideband communications
12. Rodrigues R, Mateus G, Loureiro A (1999) Optimal base station placement and fixed channel assignment applied to wireless local area network projects. In: ICON '99: proceedings of the 7th IEEE international conference on networks. IEEE computer society, Washington, DC, USA, p 186
13. de la Roche G, Rebeyrotte R, Jaffres-Runser K, Gorce J-M (2006) A QoS-based FAP criterion for indoor 802.11 wireless lan optimization. In: Proceedings of IEEE international conference on communications (ICC2006)
14. Kouhbor S, Ugon J, Rubinov A, Kruger A, Mammadov M (2006) Coverage in WLAN with minimum number of access points. In: Proceedings of IEEE 63rd vehicular technology conference, 2006. VTC 2006-Spring
15. Pabst R, Walke B, Schultz D, Herhold P, Yanikomeroglu H, Mukherjee S, Viswanathan H, Lott M, Zirwas W, Dohler M, Aghvami H, Falconer D, Fettweis G (2004) Relay-based deployment concepts for wireless and mobile broadband radio. IEEE Communications Magazine 42(9):80–89
16. Fong B, Ansari N, Fong A, Hong G, Rapajic P (2004) On the scalability of fixed broadband wireless access network deployment. IEEE Communications Magazine 42(9):12–18
17. Wang W, Wang Y, Li X-Y, Song W-Z, Frieder O (2006) Efficient interference-aware TDMA link scheduling for static wireless networks. In: 12th ACM annual international conference on mobile computing and networking (MobiCom 2006)
18. Yi S, Pei Y, Kalyanaraman S (2003) On the capacity improvement of ad hoc wireless networks using directional antennas In: Proceedings of the 4th ACM MobiHoc, pp 108–116
19. Jain K, Padhye J, Padmanabhan VN, Qiu L (2003) Impact of interference on multi-hop wireless network performance. In: MobiCom '03: proceedings of the 9th annual international conference on mobile computing and networking. ACM Press, New York NY USA, pp 66–80
20. Raniwala A, Chiueh T (2005) Architecture and algorithms for an ieee 80211-based multi-channel wireless mesh network. In: Proc of IEEE INFOCOM, 2005
21. Kyasanur P, Vaidya NH (2005) Routing and interface assignment in multi-channel multi-interface wireless networks. In: Proc of IEEE WCNC
22. Draves R, Padhye J, Zill B (2004) Routing in multi-radio multi-hop wireless mesh networks. In: Proc of ACM Mobicom
23. Wang J, Fang Y, Wu D (2006) A power-saving multi-radio multi-channel MAC protocol for wireless local area networks. In: Proc of IEEE international conference on computer communication (INFOCOM'06).
24. Chandra R, Qiu L, Jain K, Mahdian M (2004) Optimizing the placement of internet TAPs in wireless neighborhood networks. In: Proceedings of the 12th IEEE international conference on network protocols (ICNP 2004)
25. Meng X, Tan K, Zhang Q (2006) Joint routing and channel assignment in multi-radio wireless mesh networks. In: Proc of IEEE international conference on communications (ICC '06)

Fan Li received the MEng degree in electrical engineering from the University of Delaware in 2004, the MEng degree and BEng degree in communications and information system from Huazhong University of Science and Technology, China. She is currently a PhD student in the University of North Carolina at Charlotte, majoring in computer science. Her current research focuses on wireless networks, ad hoc and sensor networks, and mobile computing.

Yu Wang received the PhD degree in computer science from Illinois Institute of Technology in 2004, the BEng degree and the MEng degree in computer science from Tsinghua University, China, in 1998 and 2000. He has been an assistant professor of computer science at the University of North Carolina at Charlotte since 2004. His current research interests include wireless networks, ad hoc and sensor networks, mobile computing, and algorithm design. He has published more than 60 papers in peer-reviewed journals and conferences. He has served as program chair, publicity chair, and program committee member for several international conferences. He is the program co-chair of the first ACM International Workshop on Foundations of Wireless Ad Hoc and Sensor Networking and Computing (FOWANC 2008), and was the program co-chair of the 26th IEEE International Performance Computing and Communications Conference (IEEE IPCCC 2007). Dr. Wang is an editorial board member of the International Journal of Ad Hoc and Ubiquitous Computing, and an associate editor of the International Journal of Mobile Communications, Networks, and Computing. Dr. Wang is a recipient of Ralph E. Powe Junior Faculty Enhancement Awards from Oak Ridge Associated Universities. He is a member of the ACM, IEEE, and IEEE Communications Society.



Xiang-Yang Li has been an Associate Professor (since 2006) and Assistant Professor (from 2000 to 2006) of Computer Science at the Illinois Institute of Technology. He is a visiting professor of Microsoft Research Asia for one year from May, 2007. He also holds visiting professorship or adjunct-professorship at the following universities in China: TianJing University, WuHan University, and NanJing University. He received MS (2000) and PhD (2001) degree at Department of Computer Science from University of Illinois at Urbana-Champaign. He received the Bachelor degree at Department of Computer Science and Bachelor degree at Department of Business Management from Tsinghua University, P.R. China, both in 1995. His research interests span the wireless ad hoc networks, game theory, computational geometry, and cryptography and network security. He has published more than 120 papers in top-quality conferences and journals. He served various positions (various chairs and TPC members) at numerous international conferences. He has also been invited to serve on the panel or review research proposals such as National Science Fundation (US), National Science Fundation of China, and RGC HongKong. He is an editor of Ad Hoc & Sensor Wireless Networks: An International Journal. He is an Associate Member of the IEEE, and a member of ACM.

Ashraf Nusairat is Computer Science PhD student at Illinois Institute of Technology and works for Motorola Inc as Wireless Broadband System Architect. He received his MS in Computer Science in 1999 from The University of Akron - OH and received his Bachelor degree in Computer Engineering in 1993 from Jordan University of Science and Technology, Jordan. His research interests are in the areas of wireless broadband networks, wireless mesh networks and wireless ad-hoc networks.

Yanwei Wu is a Ph.D. candidate of Computer Science at the Illinois Institute of Technology. Her research interests are in the areas of algorithm analysis and design, wireless networks, and game theory. She has published several papers on the conferences and journals in related fields, such as throughput optimization in mesh network or wireless ad hoc network, energy efficiency in wireless network, game theory and security in wireless network. In the summer of 2007, she was an Research Aide in Argonne National lab, researching on agent based modeling.