



A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks

Mahesh K. Marina^{a,*}, Samir R. Das^b, Anand Prabhu Subramanian^b

^a School of Informatics, The University of Edinburgh, UK

^b Computer Science Department, Stony Brook University, USA

ARTICLE INFO

Article history:

Available online 5 August 2009

Keywords:

Mesh networks
Multihop wireless networks
Multiple-radio systems
Multi-channel
Channel assignment
Complexity

ABSTRACT

We consider the channel assignment problem in a multi-radio wireless mesh network that involves assigning channels to radio interfaces for achieving efficient channel utilization. We present a graph-theoretic formulation of the channel assignment guided by a novel topology control perspective, and show that the resulting optimization problem is NP-complete. We also present an ILP formulation that is used for obtaining a lower bound for the optimum. We then develop a new greedy heuristic channel assignment algorithm (termed CLICA) for finding connected, low interference topologies by utilizing multiple channels. Our evaluations show that the proposed CLICA algorithm exhibits similar behavior and comparable performance relative to the optimum bound with respect to interference and capacity measures. Moreover, our extensive simulation studies show that it can provide a large reduction in interference even with a small number of radios per node, which in turn leads to significant gains in both link layer and multihop performance in 802.11-based multi-radio mesh networks.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Wireless mesh networking is emerging as a promising technology for low-cost, ubiquitous broadband Internet access via reduced dependence on the wired infrastructure. In a wireless mesh network, a collection of stationary *wireless access routers* provide connectivity to mobile clients akin to access points in a traditional wireless LAN; but access routers communicate with each other wirelessly, potentially over multiple hops; a small fraction of those access routers are wired to the Internet and serve as Internet gateways for the rest of the network. Mesh networks based on commodity 802.11 [1] hardware and employing self-configuring ad hoc networking techniques can offer wider coverage with less expense and easier deployment. Furthermore, inherent redundancy in the mesh topology enhances reliability. Consequently, mesh networks enable a

number of new application scenarios, including community wireless networking to provide affordable Internet access especially beneficial for low-income neighborhoods and scarcely populated areas. See [2–4] for a detailed discussion on several application scenarios, and various community and commercial mesh network deployment efforts.

Multi-radio wireless mesh network architecture, in which each access router is equipped with multiple 802.11 radios, is commonly seen as a practical way for efficient utilization of available spectrum [4–20], thereby alleviating the performance degradation in multihop wireless networks arising from the need to share the wireless medium among neighboring transmissions and the ensuing multiple access interference [21,22]. Multiple channels (separated in frequency) are available with the IEEE 802.11 standard – 3 (non-overlapping) channels with IEEE 802.11b/g standards in the 2.4 GHz band and 8–24 channels with the IEEE 802.11a standard in the 5 GHz band, but the 802.11 medium access control (MAC) protocol is designed to operate over a half-duplex radio on a single

* Corresponding author.

E-mail address: mmarina@inf.ed.ac.uk (M.K. Marina).

channel. Though several solutions for using multiple channels with single radio per node¹ are available (see [23,24], for example), comparatively, multi-radio solutions operating above the MAC layer have several advantages such as being able to work with commodity 802.11 hardware without requiring MAC modifications or tight synchronization, simplifying protocol design and permitting simultaneous transmission and reception with half-duplex radios (on different channels). On the other hand, the increased energy consumption and form-factor from use of multiple radios per node are not major concerns with stationary mesh routers plugged into power outlets.

A key issue to be addressed in a multi-radio mesh network architecture and the focus of this paper is the *channel assignment* problem that involves assigning (mapping) channels to radio interfaces to achieve efficient utilization of available channels. This problem is non-trivial in the typical case where the number of radio interfaces per node is smaller in relation to the number of available channels. We can broadly classify approaches for channel assignment into two categories: *traffic-independent* and *traffic-aware*. With the traffic-independent approach, as the name suggests, channel assignment is done without explicitly considering network traffic/load. Instead it is based on physical connectivity, and a model for “potential” interference experienced/caused by a transmission or long-term interference measurements (e.g., [13–16,20]). Traffic-aware channel assignment (taken in [8–11]), on the other hand, is affected by the traffic condition in the network and therefore is based on “actual” interference. We adopt the traffic-independent approach in this work given its various practical benefits, elaborated in Section 2. Note that our approach does not preclude the option of incorporating traffic awareness in scenarios where traffic characteristics (and patterns) are known or can be predicted; specifically, traffic-awareness can be overlaid on our approach via an appropriate choice of node priorities in our channel assignment algorithm (see Section 4).

This paper makes the following contributions:

- By viewing the channel assignment as a topology control problem, we present a graph-theoretic formulation that seeks to minimize maximum interference in the network while preserving the connectivity from a single channel scenario. We show that this problem is a generalized version of the graph edge coloring problem and hence is NP-complete. We also present an integer linear program (ILP) formulation of the channel assignment problem by developing a variant of the model proposed in [13] to suit our specific optimization objective and constraints.
- We design a centralized channel assignment algorithm termed CLICA that greedily finds low interference topologies while preserving connectivity. Note that a centralized algorithm is adequate for most existing deployments given their small scale, and especially so with a fixed number of channels as is common currently. This is because channel assignments need to be updated

infrequently (in the order of tens of minutes to a few hours) in response to rare events like node failures, so its associated communication overhead (for collecting connectivity/interference information from the network at a central node and disseminating channel assignments back) is amortized. CLICA can also be used as a benchmark when evaluating distributed channel assignment algorithms. Using both graph-based and ns-2 simulations, we show that CLICA performs significantly better than a single channel and CCA (assumed in [6]), and exhibits similar behavior as optimum bounds while being comparable in terms of interference and capacity.

In comparison with other work on channel assignment taking the traffic-independent approach [6,13,15,16,20], our work offers an algorithm for finding channel-diverse and low interference topologies without sacrificing connectivity, thereby making efficient use of channels and interfaces. Our work differs from previous work in two other respects: (i) we address the computational complexity issue of the minimum interference channel assignment problem; (ii) we study maximum interference achieved by our algorithm relative to the lower bound for the optimum obtained via relaxation of our ILP model.

The rest of the paper is structured as follows: Section 2 discusses related work. In Section 3, we introduce our network model, present two different formulations of the channel assignment problem and study its computational complexity. Section 4 presents a greedy channel assignment heuristic algorithm termed CLICA to find connected and low interference topologies. In Section 5, we evaluate CLICA using a combination of graph-based simulations and packet level simulations with the ns-2 simulator. Section 6 summarizes our contributions and identifies avenues for future work.

2. Related work

2.1. Channel assignment in related domains

Channel assignment is a well studied problem in cellular networks [25], where the goal is to share channels among base stations in neighboring cells and reuse channels across distant cells. A similar problem also arises in the context of wireless LANs for allocating multiple channels among access points (APs) [26]. In both cellular networks and wireless LANs, maintaining connectivity between cells is not a concern during channel allocation as base stations and APs are interconnected by a wired backbone. In contrast, when we go to an all-wireless or multihop wireless context as with multi-radio mesh networks, connectivity needs to be kept in mind when doing channel assignment so that neighboring nodes that need to communicate are assigned a common channel. Channel assignment has also been studied in the past for general multihop wireless networks or ad hoc networks by modeling channel assignment problems as variants of graph coloring problems (see [27] and references therein). In that body of work, the optimization goal is typically to minimize the number of channels (colors) for interference-free

¹ We will use the terms “node” and “access router” synonymously.

assignment (e.g., to find a minimum length schedule in the case of TDMA scheduling). In contrast, our focus is on minimizing interference and in particular, we target scenarios with a limited number of channels (e.g., 802.11-based mesh networks).

2.2. Single radio solutions

Early work on using multiple channels in multihop wireless networks has assumed a single radio per node [23,24]. A common theme across these single-radio solutions is for each node to dynamically switch between channels, while coordinating with neighboring nodes to ensure communication over a common channel for some period. However, such coordination is usually based on tight time synchronization among nodes, which is difficult to realize in a multihop wireless network, and/or fast channel switching capability that is not yet available with commodity hardware. The MMAC protocol [23] is a variant of the 802.11 MAC protocol to allow communicating node pairs to dynamically select an appropriate operating channel for best immunity from interference. In this protocol, nodes periodically return to a common control channel to negotiate channel selection and then operate on the negotiated channel. This technique assumes tight time synchronization among nodes and fast channel switching capability. The SSCH protocol [24] is a single-radio solution which preserves the 802.11 MAC protocol, but still requires link-layer techniques for time synchronization and packet scheduling. The idea is for each node to “hop” across available channels in a manner such that communicating nodes have the same channel at the same time at some intervals giving them an opportunity to communicate directly. The SSCH technique also requires fast channel switching (in the order of 100 μ s), whereas switching delays with current commodity 802.11 hardware can be up to 100 ms [17,28].

2.3. Multi-radio solutions

Bahl et al. [5] argue that multi-radio platforms can offer significant benefits in wireless systems by discussing their use for addressing a wide range of problems such as energy conservation, capacity enhancement and mobility management. Adya et al. [6] propose a link layer protocol for multi-radio multihop wireless networks called MUP for load-aware channel selection using locally available information when a node has multiple interface/channel choices to communicate with a neighbor. This is done based on a given channel assignment for radio interfaces. In their followup work [7], a routing metric called WCETT is proposed for high throughput path selection in multi-radio multihop wireless networks that takes channel diversity into account. Both [6] and [7] assume a simple channel assignment scheme in which radio interfaces at each node are assigned to the same set of channels. We refer to this scheme as common channel assignment (CCA). Clearly, CCA is traffic-independent and leads to inefficient channel utilization in the typical case where the number of interfaces per node is smaller in relation to the number of channels.

Broadly speaking, channel assignment in multi-radio multihop wireless networks can be classified into two categories: *traffic-independent* and *traffic-aware*. In the traffic-independent approach, as the name suggests, channel assignment is done without explicitly considering network traffic/load (e.g., CCA). Instead it is based on physical connectivity, and often additionally based on a model for “potential” interference experienced/caused by a transmission or long-term interference measurements (e.g., [13–16,20]). This approach clearly suggests that routing and channel assignment functions are carried out separately with routing dependent on the result of channel assignment. Traffic-aware channel assignment, on the other hand, is affected by the traffic condition in the network and therefore is based on “actual” interference. Traffic-aware channel assignment can be done separately but iteratively with routing as in [8], or alternatively as part of a joint optimization of routing, channel assignment and scheduling as in [9–11]. Relatively speaking, the traffic independent approach has certain practical benefits even though traffic-aware channel assignment is necessary from an optimality standpoint. These benefits include: (i) stable operation, and allowing modular operation in keeping with the traditional layered protocol architecture such as using any routing protocol [20]; (ii) potentially low overhead as the channel assignment needs to be redone infrequently; (iii) obviate the need for fast channel switching, which is not currently feasible with commodity 802.11 hardware; (iv) can be the basis for more adaptive assignments as discussed in an earlier version of this paper [14] (e.g., as a base assignment that can be used for subsequent coordination when reassigning channels to interfaces or as part of a hybrid channel assignment strategy). In view of the above discussion, we adopt the traffic-independent approach in this work. Note that our approach does not preclude the option of incorporating traffic awareness in scenarios where traffic characteristics (and patterns) are known or can be predicted; specifically, traffic-awareness can be overlaid on our approach via an appropriate choice of node priorities in our channel assignment algorithm (see Section 4). In the rest of this section, we review the previous work on traffic-aware and traffic-independent channel assignment for multi-radio mesh networks.

2.3.1. Traffic-aware channel assignment

Raniwala et al. [8] take a traffic-aware approach to address the combined channel assignment and routing problem in a centralized fashion based on *a priori* knowledge of the traffic profile. Their solution iteratively applies load-balanced routing and channel assignment heuristic until capacity assigned to each link exceeds the expected load on that link. Alicherry et al. [9] propose a constant factor approximation algorithm for joint routing, channel assignment and scheduling to optimize overall network throughput subject to fairness constraints. Kodialam and Nandagopal [10] address the joint optimization of routing, channel assignment and scheduling to get upper and lower bounds on capacity (along the lines of [22]) for the problem of feasibility of a given end-to-end traffic demand vector. Unlike [9], this work considers a general network model and channel switching. Zhang et al. [11] use a col-

umn generation approach to solve the joint optimization of routing, channel assignment and scheduling to minimize system activation time to satisfy given end-to-end traffic demands; this work allows channel switching as in [10]. The above three pieces of work [9–11] assume that the system operates in a synchronous time-slotted mode, thus do not accurately capture asynchronous common multi-radio multihop wireless networks (based on 802.11). Kyasanur and Vaidya [12] extend the work of Gupta and Kumar [21] to include the case when the number of radio interfaces per node is less than the number of available channels and derive corresponding asymptotic capacity bounds. An observation from this work is of practical interest, that is: one interface is sufficient in random networks to achieve the capacity bound when network size in terms of number of nodes (N) is much greater than the number of channels (up to $O(\log N)$) and there is no switching delay, but more than one interface is needed in presence of switching delay. They also point out that protocol design is simplified from the use of multiple interfaces.

2.3.2. Traffic-independent channel assignment

Roy and co-workers [13] consider multi-radio channel assignment with the goal of maximizing the number of links active concurrently (in other words, the link layer capacity) and present two alternative ILP models, but do not discuss any algorithm for channel assignment. We compare the link layer capacity with our algorithm to the bound obtained using one of the ILP models in [13]. We also develop a variant of that ILP model to reflect our optimization goal (minimizing interference) and various constraints. Tang and co-workers [15] present a centralized heuristic channel assignment algorithm to obtain a K-connected topology that minimizes maximum interference (as in our case). However, they do not address the computational complexity issue nor do they compare their algorithm with (bound for) the optimum. Ramachandran et al. [16] propose a multi-radio mesh architecture where each node dedicates one interface at each node to operate over a network-wide common (default) channel, while a centralized channel assignment algorithm is used to assign channels to remaining interfaces; default channel selection accounts for interference from co-located wireless networks. Having all nodes use one of their interfaces to operate over a common channel can lead to inefficient utilization of channels and interfaces in the common case with fewer than a handful of interfaces per node. On the other hand, when common channel assumption is dropped, their channel assignment algorithm can cause network partitions. As with [15], computational complexity issue and characterizing algorithm performance with respect to the optimum are not addressed in [16].

2.3.3. Channel assignment protocols

Some distributed channel assignment algorithms exist [17–20], but all of them rely on some structure to permit distributed operation. Raniwala and Chiueh [17] propose distributed and traffic-aware channel assignment and routing algorithms assuming a tree-structure, which is optimized for a specific traffic pattern. Zhu and Roy [18] propose a clustered multi-channel two-radio (CMT) archi-

ture in which each node is assumed to have two interfaces (referred to as default and secondary, respectively) and nodes organize themselves into clusters; the default interface is used for inter-cluster communication on a common channel, whereas secondary interface is used for intra-cluster communication on a channel chosen by the corresponding clusterhead to minimize interference with neighboring clusters. This architecture cannot exploit additional interfaces (beyond two) because of the design assumption; besides, it leads to poor utilization of channels since all default interfaces use the same channel. Kyasanur and Vaidya [19] propose a hybrid channel assignment strategy where each node assigns a fixed channel to one of its interfaces to receive data from neighboring nodes, while using its other interfaces with dynamic channel switching to transmit data to neighbors. Depending on the traffic pattern, a single interface for reception can become a bottleneck. Both [18] and [19] can be viewed as hybrid or partially traffic-aware techniques as they use fixed channels for some of the interfaces. Ko et al. [20] propose a distributed and traffic-independent channel assignment algorithm with the simplifying assumption that all nodes use one of their interfaces to operate over a common channel (as in [16]).

3. Model and problem formulation

We consider a wireless mesh network formed by a set of stationary access routers (nodes). Each node is equipped with one or more radio interfaces for backhaul communication, i.e., communication with other nodes in the mesh network. We assume that all such radio interfaces are half-duplex, use omni-directional antennas and have identical communication ranges (denoted by R). We model the connectivity between mesh nodes by an undirected graph $C = (\mathcal{N}, \mathcal{L})$, henceforth referred to as the *connectivity graph*. Here \mathcal{N} denotes the set of nodes, whereas \mathcal{L} denotes the set of links. A pair of nodes have a link in \mathcal{L} , if they are physically located within each other's communication range.

Suppose that there are M distinct wireless channels denoted by c_1, c_2, \dots, c_M . Let I_i denote the number of radio interfaces at node i available for backhaul mesh communication, where $1 \leq I_i \leq M$. Note that different nodes may have a different number of radio interfaces. The assignment of channels to radios induces the *network topology*, T . The network topology in general is an undirected multigraph without self-loops. It may not be identical to the connectivity graph, C . This can happen because of two reasons. First, a link in C may be absent in T if the nodes at the end points of this link do not have any radios assigned to the same channel. Second, it may have several corresponding links in T if the nodes at the end points have more than one radio each with common channels. *Our goal in this paper is to seek channel assignments which ensure that C is a spanning subgraph of T while reducing interference on any given channel.*

We assume that each of the mesh nodes is equipped with AP functionality to serve mobile clients (users) and that mesh nodes communicate with mobile clients over

an additional radio interface in a separate band. For instance, as noted in [4], it is common to use 5 GHz band for mesh backhaul communication and 2.4 GHz band for client access. Mesh nodes acting as Internet gateways additionally have a wired interface.

3.1. Modeling interference

The broadcast nature of the wireless medium makes it crucial to account for multiple access interference. The success of a transmission is dependent on the extent of wireless interference in the vicinity of the transmission.

Let us first consider the single wireless channel case. Two models are commonly used for accounting the impact of interference on the success of a given transmission [22,21]: the *protocol model* and the *Physical Model*. Using these models, we can infer the *potential* interference between a pair of transmissions (links). We focus on the simpler protocol model in this paper for ease of exposition, but it is straightforward to extend our discussion to the more sophisticated physical model (see Appendix).

3.1.1. Protocol model

This model associates an interference range for each node (typically, larger than the communication range) that defines the range up to which a transmitter can interfere with the reception at an unintended receiver. Suppose that all nodes have identical interference ranges (denoted by $R' \geq R$). According to the protocol model, a transmission from node i to node j is successful provided no other node located within a distance R' from j transmits at the same time. For reliable unicast transmission (e.g., 802.11 RTS/CTS/DATA/ACK exchange) transmissions in both directions must be successful. Thus, it is additionally required that all nodes located within R' from i refrain from transmitting as well. Note that the protocol model naturally models half-duplex radios. Also note that the model presented here is a slight variation of the original protocol model in that we assume identical communication and interference ranges as well as consider bidirectional (DATA-ACK) communication as in 802.11.

The next level of modeling interference is to represent interference among all possible transmissions in the network. The *conflict graph* framework proposed in [22] offers a flexible and fine-grained approach for this purpose. The conflict graph, $CG = (\mathcal{V}_{cg}, \mathcal{E}_{cg})$, contains a vertex (denoted by l_{ij}) corresponding to every link $i-j$ in the network topology between nodes i and j . We place an edge² between two nodes (say, l_{ij} and l_{pq}) in the conflict graph if the corresponding links ($i-j$ and $p-q$) in the network can interfere. The conflict graph in general is *weighted* with the weight of an edge indicating the extent of interference between the vertices of that edge (corresponding to two links in the network topology). For the protocol model, the conflict graph contains an edge between two vertices (l_{ij} and l_{pq}) if either nodes i or j are located within distance R' from p or q ; the conflict graph is undirected and all edges have

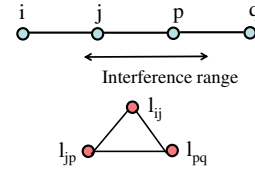


Fig. 1. A four node network (above) and the corresponding conflict graph based on the protocol model (below). The edge weights in the conflict graph are unity.

unit weight in this case. See Fig. 1 for an illustration. In the appendix, we describe the physical model and a method to find the edge weights for the conflict graph with that model, based on [22].

With multiple channels, interference modeling depends on the inter-channel interference in addition to interference within each channel (i.e., co-channel or intra-channel interference), which is modeled as in the single channel case described above. For non-interfering channels (as would typically be the case if we only consider non-overlapping channels), we can consider each channel and links using that channel *independently* and obtain a corresponding conflict graph (as in the single channel case); the overall conflict graph then is a union of conflict graphs for each individual channel. When different channels can interfere, such as with a partially overlapping set of channels, we also need to take that interference into account. The concept of interference-factor (I-factor) proposed in [26] can be used for this purpose.

3.2. Topology control perspective and graph-theoretic formulation

The channel assignment problem in a multi-radio mesh network involves obtaining a mapping between radios and channels. We view it as a topology control problem. Unlike a wired network, links in a wireless network are flexible entities which can be configured or tuned [29]. There are a wide variety of tunable link parameters including transmission power, bit rate, frequency band/channel and beam direction (if directional antennas are used). In a broad sense, topology control is a way to exploit such link controllability to obtain a desired topology. Topology control is typically targeted towards reducing interference, or improving energy efficiency while maintaining network connectivity [30]. Our focus here is on reducing interference. While much of the topology control work is aimed at transmit power control [31,30], we look at *multi-radio channel assignment as yet another way to perform topology control*. Our main goal is to reduce the interference on any given channel by distributing it across as many channels as possible, thereby facilitating effective utilization of the available channels.

Generally speaking, the “goodness” of a channel assignment rests on two factors: connectivity and interference. In a multi-radio multihop wireless network with multiple available channels, two nodes can communicate only if each of them has a radio interface assigned to a common channel. Assigning many interfaces in the network to a

² As in [22], we associate the terms “node” and “link” with the network topology and the connectivity graph, and use the terms “vertex” and “edge” for the conflict graph.

few channels can provide richer connectivity, but it has the undesirable effect of increasing interference among transmissions on those channels. Thus, the channel assignment has to balance between minimizing interference (on any given channel) and maintaining sufficient connectivity. *It is this need to balance between interference reduction and connectivity maintenance that makes the multi-radio channel assignment a topology control problem.* With the above view, channel assignment becomes an optimization problem, where “some interference measure” defined over the whole network according to a given interference model is optimized, with the constraint that “some notion of connectivity” is preserved. To define the problem more concretely, we discuss below our specific choices for the optimization (interference reduction) criterion, the connectivity constraint and the interference model.

Before we discuss our objective function, we introduce a basic interference measure called *link conflict weight* used in our optimization. Link conflict weight of a link in the network topology is defined as the sum of weights of edges incident on the corresponding vertex in the conflict graph. For the link $i-j$ assigned to channel c , the link conflict weight is denoted by $W(l_{ij}^c)$. We seek to minimize the maximum interference (i.e., link conflict weight) over all channels in the resultant network topology. This objective leads to a more even spreading of interference across all available channels. A similar objective has been considered in the past in other contexts [30,26].

For the connectivity constraint, as in [13], we require that all links in the connectivity graph are still “preserved” in the network topology after the channel assignment is complete, that is if two nodes are within the communication range of each other, then they are assigned a common channel. In other words, network connectivity in the single channel case is maintained even after channel assignment and therefore, path characteristics such as shortest path length (in number of hops) do not get worse due to channel assignment. Even though the specific connectivity constraint we place may seem quite restrictive, it does not limit the potential topology control gains. In fact, as will be evident later, our approach can be seen as a way of making the conflict (interference) graph sparser through the use of multiple channels – a sparser conflict graph implies reduced interference.

For the interference model, we use the protocol model (described in the previous subsection) and assume non-interfering channels. Nevertheless, extensions to other interference models (e.g., the physical model) and interfering channels are straightforward given the flexible nature of the conflict graph approach we adopt to represent interference.

We refer to the decision problem equivalent of the above optimization problem as *Connectivity-preserving Interference-bounded Channel Assignment*. The decision version can be stated as follows.

INSTANCE: Connectivity graph $C = (\mathcal{N}, \mathcal{L})$; M distinct channels; I_i backhaul radio interfaces at node i ; Interference model; non-negative integer B .

QUESTION: Is there a connectivity-preserving assignment of channels to radios such that the maximum link conflict weight in the resultant network topology $\leq B$?

3.3. Analysis of complexity

Most of the traditional channel assignment problems for wireless networks are known to be difficult and have a close relationship with graph coloring problems [27,25]. Even though the connection between these problems and the above channel assignment problem may not be apparent, a closer look does reveal that our problem is in fact a generalized version of a well-known graph edge coloring problem. Unlike these other problems which seek proper coloring (i.e., conflict-free channel assignment), we attempt to minimize a measure of conflict. Though the extreme cases (where there is either one radio per node or as many radios as the number of channels) are trivial, the general problem is intractable. Below, we show that it belongs to the class of NP-complete problems.

Theorem 1. *The connectivity-preserving interference-bounded channel assignment problem as stated above is NP-complete.*

Proof. The problem is clearly in NP since an assignment can be verified in polynomial time.

The rest of the proof is by restriction [32]. We show that the above channel assignment problem contains a known NP-complete problem *minimum edge coloring* (also called *minimum chromatic index*) [33] as a special case. For clarity, *minimum edge coloring* is restated below from [32].

INSTANCE: Graph $G = (\mathcal{V}, \mathcal{E})$ and a positive integer K .

QUESTION: Does G have a chromatic index $\leq K$, i.e., can \mathcal{E} be partitioned into disjoint sets $\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_k$ (each set denoting a particular color, or in our case, channel), with $k \leq K$, such that, for $1 \leq i \leq k$, no two edges in \mathcal{E}_i share a common end point in G ?

Now we note that a specific instance of the *connectivity-preserving interference-bounded channel assignment* problem is identical to the *minimum edge coloring* problem. The following conditions hold for this instance: (i) C and G are identical; (ii) $M = K =$ maximum node degree in C ; (iii) $I_i =$ degree of node i ; (iii) a simple one-hop interference model holds, where two links in \mathcal{L} interfere only if they are assigned the same channel (i.e., have the same color) and share a common end point in C ; (iv) $B = 0$. \square

The problem remains NP-complete even when the number of channels, $M = 3$. So, the difficulty does not arise from having more channels. The problem is also strongly NP-complete. Using a similar proof as above, we can show that the problem of minimizing average interference is also NP-complete.

3.4. Integer linear program (ILP) formulation

Given that our channel assignment problem is NP-hard (see previous subsection), our focus in this paper is to develop a good heuristic algorithm that finds connected and low interference topologies, which is presented in Section 4. Ideally, we would like to compare our channel assignment algorithm to the optimum solution. Since it is prohibitively time-consuming to compute the optimum solution, we instead focus on obtaining a lower bound of the optimum. With this intention in mind, in this subsection we

formulate our channel assignment problem as an integer linear program (ILP) and derive a lower bound via its relaxation to an LP problem (i.e., omitting the integrality requirement of the variables in the ILP).

The ILP formulation assuming the protocol interference model and non-interfering channels is detailed below. We first introduce additional terminology and definitions:

- \mathcal{M} denotes the set of available non-interfering channels.
- CG' denotes the adjacency matrix for the conflict graph corresponding to the connectivity graph, C .
- The variable X represents the weights of edges in the conflict graph.
- The variable CA is a channel assignment matrix indicating whether a link l in the connectivity graph, C is assigned a channel, m .
- $rows(n)$, like in [13], is defined as the set of row indices in the channel assignment matrix, CA , such that n is an end node in the links corresponding to $rows(n)$.
- The variable Y , also defined as in [13], is a $N \times M$ matrix such that $Y_{nm} = 1$ if at least one link incident on node n has been assigned a channel m .

The ILP is given below.

Minimize f
subject to

Interference Constraint:

$$f \geq \sum_{b \in \mathcal{L}} X_{ab}^m; \quad \forall a \in \mathcal{L}, \forall m \in \mathcal{M} \quad \text{such that } CG'_{ab} = 1 \quad (3.1)$$

$$X_{ab}^m \geq CA_{am} + CA_{bm} - 1; \quad \forall a \in \mathcal{L}, \forall b \in \mathcal{L}, \forall m \in \mathcal{M} \quad \text{such that } CG'_{ab} = 1 \quad (3.2)$$

Connectivity Constraint:

$$\sum_{m \in \mathcal{M}} CA_{lm} \geq 1; \quad \forall l \in \mathcal{L} \quad (3.3)$$

$$\sum_{m \in \mathcal{M}} CA_{lm} \leq M; \quad \forall l \in \mathcal{L} \quad (3.4)$$

Interface Constraint:

$$CA_{jm} - Y_{nm} \leq 0; \quad j \in rows(n), \forall n \in \mathcal{N}, \forall m \in \mathcal{M} \quad (3.5)$$

$$\sum_{m \in \mathcal{M}} Y_{nm} \leq I_n; \quad \forall n \in \mathcal{N} \quad (3.6)$$

Variable-Type Constraints:

$$X_{ab}^m \in \{0, 1\}; \quad \forall a \in \mathcal{L}, \forall b \in \mathcal{L}, \forall m \in \mathcal{M} \quad (3.7)$$

$$CA_{lm} \in \{0, 1\}; \quad \forall l \in \mathcal{L}, \forall m \in \mathcal{M} \quad (3.8)$$

$$Y_{nm} \in \{0, 1\}; \quad \forall n \in \mathcal{N}, \forall m \in \mathcal{M} \quad (3.9)$$

Recall that our objective is to minimize the maximum interference. Since it is a non-linear objective, we linearize it based on the principle described in Rardin [34], which results in the addition of a new constraint (3.1). The main constraints in the formulation are divided into three categories: interference, connectivity and interface constraints, respectively. Connectivity constraints in (3.3) and (3.4) ensure that each link in the connectivity graph is assigned at least one channel. Interface constraints ((3.5) and (3.6)) modeled as in [13] ensure that the number of channels as-

signed to each node are limited to the number of interfaces available at that node.

To obtain the lower bound, we relax the above ILP by dropping the integrality requirement for variables X_{ab}^m , CA_{lm} and Y_{nm} in constraints (3.7)–(3.9) so that they can take any real value between 0 and 1. In order to get a tighter lower bound, we also include two additional constraints in the relaxed LP as discussed below based on the work in [35].

For each vertex u in \mathcal{V}_{cg} , we compute a *maximal* clique containing the vertex u using a simple greedy approach. For each vertex u , let S_u be the set of vertices in the maximal clique thus computed. It can be shown that the number of edges given the same color in the complete subgraph of size $|S_u|$ when colored by M colors is at least $\sigma(S_u, M)$ [35], which is given by

$$\sigma(S_u, M) = \frac{\beta\alpha(\alpha + 1) + (M - \beta\alpha(\alpha - 1))}{2}$$

where $\alpha = \lfloor \frac{|S_u|}{M} \rfloor$ and $\beta = |S_u| \bmod M$.

The above observation yields the following constraint for our relaxed LP.

$$\sum_{i,j \in S_u} X_{ij} \geq \sigma(S_u, M) \quad \forall u \in \mathcal{V}_{cg} \quad (3.10)$$

We also generate another set of constraints based on the above observation as follows. Note that $rows(i)$ is the set of vertices in \mathcal{V}_{cg} corresponding to links incident on a node $i \in \mathcal{N}$. Since the set of vertices $rows(i)$ in \mathcal{V}_{cg} forms a clique in CG and uses at most I_i colors (due to the interface constraint on node i), we have the following constraint:

$$\sum_{u,v \in rows(i)} X_{uv} \geq \sigma(rows(i), I_i) \quad \forall i \in \mathcal{N} \quad (3.11)$$

4. CLICA algorithm

In this section, we develop a polynomial-time heuristic called *Connected Low Interference Channel Assignment* or CLICA for assigning channels to radios. Based on our discussion in the previous section about the close relationship between channel assignment and graph coloring, we sometimes use the term “color” in place of “channel” henceforth for ease of exposition.

Before we present our algorithm, let us first look at the pitfall of arbitrarily coloring radios and links using a simple example. Fig. 2 shows a 4 node connectivity graph with one radio per node. Suppose we are given 2 colors: c_1 and c_2 . If we first color the link $a-b$ with c_1 (by assigning channel c_1 to the radios at a and b) and later color the link $c-d$ with c_2 in a similar fashion, then we end up with a partitioned network where nodes a and b are disconnected from nodes c and d . This simple example shows that a coloring decision constrains the flexibility for future coloring decisions if we want to preserve the network connectivity. For example, nodes c and d are precluded from using color c_2 because of an earlier choice to use color c_1 for link $a-b$. However, adding more radio interfaces to nodes will provide more flexibility in coloring.

The central idea in the CLICA algorithm is to use that degree of flexibility as a guide in determining the *order* of fu-

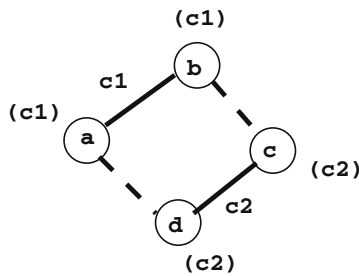


Fig. 2. Example illustrating how the connectivity constraint limits coloring choices.

ture coloring decisions. Specifically, each node is associated with a priority, and coloring decisions are made on a node-by-node basis in the order of this priority. The set of coloring decisions at a node i include choosing colors for radios at i and its adjacent nodes in order to color all links incident to i in the connectivity graph. At the beginning of the algorithm, each node is given a priority based on some criterion (e.g., closeness to wired gateway to Internet, traffic load). These priorities determine the default order for making coloring decisions. However, the algorithm, in the midst of its execution, may override that order by setting priority of a subset of nodes to a value greater than the maximum priority (over all nodes) to reflect the lack of flexibility for coloring radios at nodes in that subset. This characteristic of the CLICA algorithm to alter a node's priority during the course of its execution makes it an *adaptive priority algorithm* [36]. Here, we present CLICA in a generic form. Depending on the specific criteria used to determine the initial node priorities, specific heuristics can be realized.³

Going back to the example in Fig. 2, suppose that initial order of priorities is a, d, c and b . So, CLICA starts at a to color its incident links. Suppose it chooses c_1 to color the link $a-b$. As a result, both a and b lose further flexibility in choosing colors for their other incident links. So, CLICA additionally bumps b 's priority to the highest. Moreover, it recursively starts coloring at b to retain links on other paths connecting a and b (only one path in this example: $b-c-d-a$), which results in node b reusing color c_1 for link $b-c$. The same procedure as above (i.e., priority increase followed by recursive color reuse) repeats itself at node c forcing link $c-d$ to use c_1 , which in turn increases the priority of d . At d , since there is already a common color (c_1) with node a , the link $a-d$ is colored with c_1 . At this point, CLICA comes out of recursion and terminates. Now suppose that nodes a and d have two radios and the algorithm starts like before at a by coloring link $a-b$ with c_1 . Even in this case, the algorithm goes through recursion to color b and c ahead of d ; however unlike in the previous case the

algorithm colors the link $a-d$ with c_2 by using the additional radios. The above two cases are distinguished by lines 21 and 22 in the CLICA pseudocode shown in Fig. 3. Note that CLICA is naturally *recursive* and follows a chain of the least flexible nodes to maintain network connectivity. Also note that it is a *one-pass* algorithm in that coloring decisions once made are not reversed later in the algorithm execution.

Each coloring decision is made in a *greedy* fashion: node i , when faced with a decision to pick a color for an incident link $i-j$, makes a locally optimal choice from among the feasible set of colors: the color that minimizes the maximum link conflict weight over all links that can interfere with the link $i-j$ including itself (GreedyMax). Alternatively, we can pick the color that minimizes the link conflict weight for the link $i-j$ (GreedyAvg). Simulation results in this paper use the former method (i.e., GreedyMax).

The algorithm described so far (see Fig. 3) attempts to color all links in the connectivity graph to lower overall interference, while satisfying the connectivity constraint (see Section 3). When the algorithm terminates, it is possible that radios at two neighboring nodes share more than one common color. We represent this by adding multiple links in the network topology between such nodes. This augmented graph represents the resultant network topology (denoted by T). Also there may still remain some nodes with uncolored radios because each of those nodes have more radios than their respective degree, but coloring them can increase the potential interference of the network topology.

Now we show that CLICA indeed preserves connectivity.

Theorem 2. *CLICA algorithm yields a connectivity preserving channel assignment.*

Proof. Recall that a connectivity preserving assignment implies that for every link in the connectivity graph, radios at the end nodes are assigned at least one common channel (color).

Let us consider an arbitrary link $\langle u, v \rangle$. Without loss of generality, suppose that node u is assigned a higher priority than node v at the start of the algorithm.

Consider the case when u is visited before v (via a call to the procedure CLICA-VISIT-NODE in line 4 of ALGORITHM CLICA). There are two sub-cases within this case. In the first sub-case, u colors the link $\langle u, v \rangle$. Here we claim that v has at least one uncolored radio (see line 15 in CLICA-VISIT-NODE). Otherwise, if all radios at v are already colored (because of previous calls to CLICA-VISIT-NODE and execution of lines 7, 15) then v would have been visited prior to u via a call to CLICA-VISIT-NODE (see lines 10, 21–22), a contradiction. Thus, both u and v have will end up having a radio with a common color that matches the color of the link $\langle u, v \rangle$. In the second sub-case, v colors the link $\langle u, v \rangle$. This could happen in the following scenario. Node u colors its own last uncolored radio and that of a neighboring node w (see lines 20–21). This in turn may trigger further reuse of that color recursively among a set of nodes (including v) starting from w to preserve links on all paths between those nodes (see Fig. 4 and lines 5–10).

³ We experimented with various possibilities, but for simulation results presented in this paper, we assign initial node priorities as follows: a node is randomly chosen and assigned the highest priority; other nodes are assigned lower priorities determined by the order in which they are discovered during a depth-first search of the connectivity graph starting from the highest priority node – a node i is assigned a higher priority than a node j if i is discovered earlier than j .

ALGORITHM CLICA

Input: (1) Connectivity graph $C = (\mathcal{N}, \mathcal{L})$,
 (2) M distinct channels $\langle c_1, c_2, \dots, c_M \rangle$,
 (3) $\forall i \in \mathcal{N}, I_i (1 \leq I_i \leq M)$ radio interfaces $\langle r_1^i, r_2^i, \dots, r_{I_i}^i \rangle$,
 (4) Interference model.
Output: Assignment of channels to radio interfaces at each node that preserves the connectivity specified by C .
 This assignment determines the actual network topology.

▷ Color all links in C .

```

1  Assign a priority to each node in  $C$  based on some criterion
2  Order nodes in non-increasing order of their priorities
3  for each node  $v$  in this order
4    do CLICA-VISIT-NODE( $v, \emptyset, \text{NIL}$ )

```

PROCEDURE CLICA-VISIT-NODE($v, \text{NodeSet}, c'$)

```

1  for each uncolored link  $\langle v, w \rangle$  in  $C$ 
2    do if each of the nodes  $v$  and  $w$  have a radio with a common color “ $c$ ”
3      then color the link  $\langle v, w \rangle$  with  $c$ 
4      update conflict graph
5  if  $\text{NodeSet} \neq \emptyset$  and  $c' \neq \text{NIL}$ 
6    then while  $\exists$  an uncolored path from  $v$  to a node  $x$  in  $\text{NodeSet}$  via a neighbor  $w$ 
7      such that each intermediate node on the path has only one uncolored radio
8        do assign  $c'$  to the uncolored radio at  $w$ 
9        color the link  $\langle v, w \rangle$  with  $c'$ 
10       update conflict graph
11       CLICA-VISIT-NODE( $w, \text{NodeSet} \cup \{v\}, c'$ )
12  while  $v$  has an uncolored incident link  $\langle v, w \rangle$  in  $C$ 
13    do pick a color “ $c$ ” greedily based on objective function evaluated on the conflict graph
14    ▷ If all radios at  $v$  are already colored, then  $c$  is chosen from among currently assigned colors.
15    ▷ Otherwise, it is chosen from among unused colors.
16    if not all radios at  $v$  are colored
17      then assign  $c$  to an uncolored radio at  $v$ 
18      assign  $c$  to an uncolored radio at  $w$ 
19      color the link  $\langle v, w \rangle$  with  $c$ 
20      update conflict graph
21      if all radios at  $w$  are colored
22        then
23          ▷ This recursive step indicates implicit increase in priority of node  $w$ .
24        if all radios at  $v$  are colored
25          then CLICA-VISIT-NODE( $w, \{v\}, c$ )
26        else CLICA-VISIT-NODE( $w, \emptyset, \text{NIL}$ )

```

Fig. 3. Pseudo-code of the CLICA algorithm. The algorithm visits the nodes in the connectivity graph and chooses a color (channel) for each radio on the nodes so the original connectivity is preserved. The algorithm maintains the conflict graph on the side, which is used to model interference and guide the greedy heuristic choice of colors.

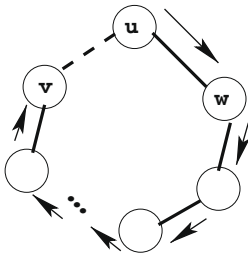


Fig. 4. Illustration helpful for proving the connectivity preservation aspect of the CLICA algorithm. See description in text.

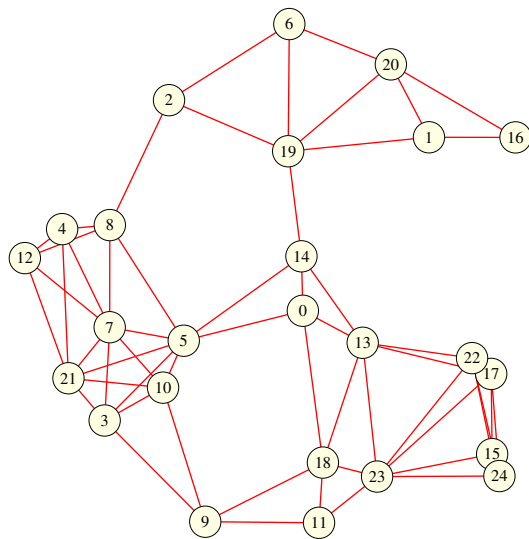
Because of the recursive color reuse, node v ends up coloring the link $\langle u, v \rangle$ with a color common to both u and v (see lines 1–4 in CLICA-VISIT-NODE).

Alternatively, v can be visited prior to u . This can happen because of a recursive call to CLICA-VISIT-NODE in lines 10, 21–22. Similar arguments as above apply in this case with roles of u and v interchanged. \square

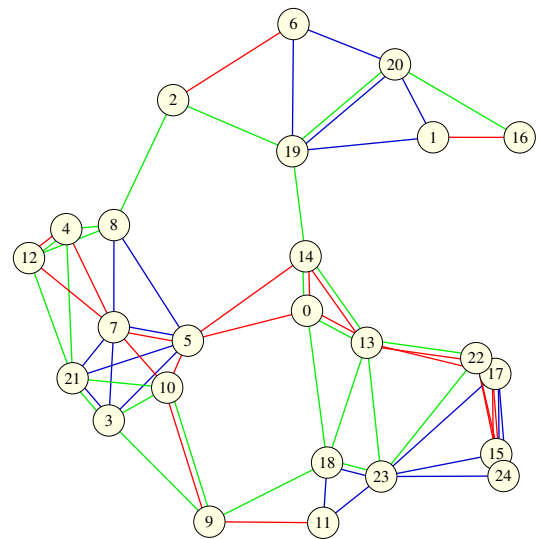
Note that CLICA is a polynomial time algorithm. While this is largely evident from the pseudo code of the algorithm given in Fig. 3, we make two important observations to reinforce this point. First, each link in the connectivity graph is only colored once by the algorithm and it terminates when all links are assigned a color. Second, the line 6 in the pseudo code can also be executed in polynomial time as we only need to search among node disjoint paths (no more than the degree of node v) and each such search can be done in linear time using a constrained version of depth-first search (DFS). Note that this line is executed via the recursive step in line 21 to “close” constrained cycles with only one uncolored radio at intermediate nodes.

Fig. 5 graphically illustrates the topologies generated by CLICA.

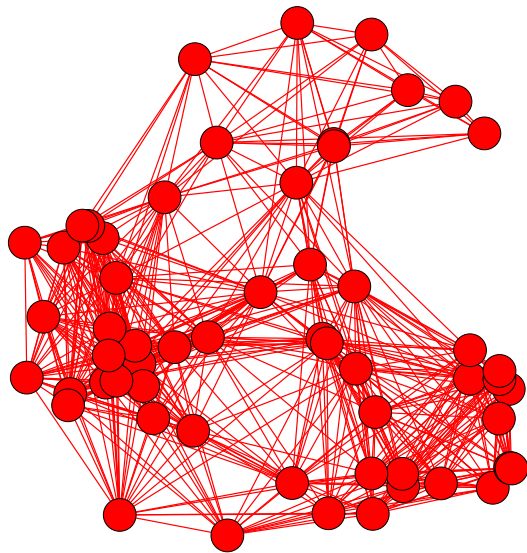
Although our description of the algorithm assumes a centralized setting, it is possible to implement a specific instance of the algorithm in a distributed manner (albeit with limited scalability). Specifically, the idea is to explore the connectivity graph via a distributed depth-first search like procedure (similar to [37]) with token-passing starting



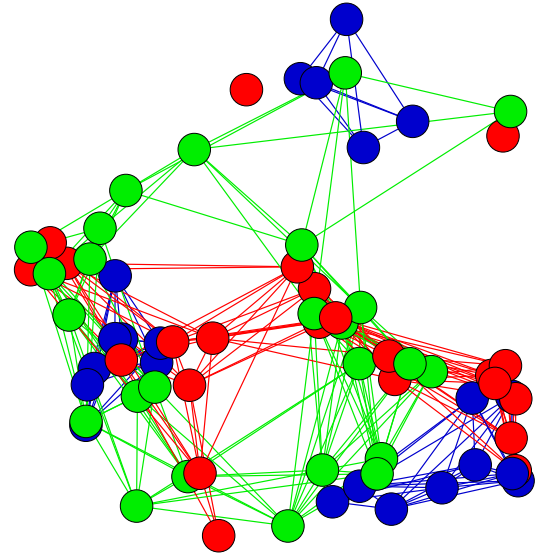
(a) Network topology (single channel)



(b) Network topology (3 channels, 2 radios per node)



(c) Conflict graph (single channel)



(d) Conflict graph (3 channels, 2 radios per node)

Fig. 5. Example showing the ability of the CLICA algorithm to generate connected and low interference topologies. This scenario corresponds to 25 randomly distributed nodes with 150 m communication range in a $500\text{ m} \times 500\text{ m}$ field, protocol interference model with identical communication and interference ranges, and non-interfering channels. Note that the interference (maximum link conflict weight) for the multi-channel case in this example is reduced by a factor of 3 relative to the single channel with only 2 radios.

from a designated node (e.g., a gateway node wired to Internet). Also note that in such an implementation each node makes coloring decisions based on its own “view” of the conflict graph, which can lead to coloring somewhat worse than a centralized solution. We will leave further investigation of this issue for the future.

5. Evaluation

In this section, we study the average-case performance of the CLICA algorithm using simulations under a varying

number of channels and radio interfaces per node. Our evaluation uses a combination of graph-based simulations and ns-2 [38] simulations.

Graph-based simulations compare interference and capacity properties of topologies generated by different channel assignment algorithms independent of protocol overheads and interactions – no protocol is modeled in these simulations. As a measure of network-wide interference, we use the maximum link conflict weight metric discussed earlier. Additionally, we use the maximum number of concurrent transmissions (calculated by computing

maximum independent set in the conflict graph) as a measure of the total one-hop capacity. Since the maximum independent set problem is itself NP-complete, we use a greedy $O(1)$ approximation algorithm mentioned in [39]. As a result, the capacity metric reported is an *underestimate* of the actual (one-hop) network capacity.

Detailed ns-2 simulations are used to evaluate CLICA performance in a 802.11-based multi-radio mesh network using the standard performance metrics of aggregate throughput and average delay.

Throughout we consider networks of 50 randomly placed nodes with 250 m communication range in a $1000\text{ m} \times 1000\text{ m}$ field. These parameters result in *connected* scenarios with high probability based on the analysis in [40]. Such topologies can be seen as the result of applying transmit power control to reduce overall interference in a single channel network. Consequently, they are a good basis to evaluate additional interference reduction possible via intelligent channel assignment. When evaluating topologies resulting from channel assignment, we also study the impact of density by additionally considering *biconnected* scenarios obtained from using a smaller field size ($850\text{ m} \times 850\text{ m}$).

We assume non-interfering channels. For interference within a channel, we assume the protocol model [21] with 550 m interference range.

The single channel case serves as a baseline in all our comparisons. We also consider the common channel assignment (CCA) algorithm used in [6], which is an alternative traffic-independent scheme from the literature. As mentioned earlier, CCA assigns channels to radio interfaces

at all nodes identically. More precisely, using the notation from our model, CCA assigns the j th interface at a node i ($1 \leq j \leq RI_i$) to the j th channel. We also compare the interference and capacity properties of topologies obtained from CLICA with a corresponding (lower or upper) bound for the optimum. For the interference (maximum link conflict weight objective), we use the lower bound derived from relaxation of the ILP presented in Section 3.4. For the capacity (maximum number of concurrent transmissions), we use the upper bound obtained via relaxation of the ILP model ‘B’ from [13]. We compute these bounds using the GNU Linear Programming Kit (GLPK).

5.1. Topology properties

To study topology properties, we generate a large number of random multihop wireless network scenarios (one thousand for each data point) with a varying number of channels and number of radios per node, and apply different channel assignment algorithms and compute bounds on them. All nodes have the same number of radios. Results for 12 channels are shown in Fig. 6; results with a different number of channels (not included for brevity) exhibit similar qualitative behavior. For CLICA, we have experimented with two types of initial node priorities: DFS-based ordering and random. We found DFS-based ordering to be consistently superior, so only show results using that approach.

CCA interference (maximum link conflict weight) performance (Fig. 6 (a, b)) is unaffected by the number of channels and radios per node, whereas its capacity (maxi-

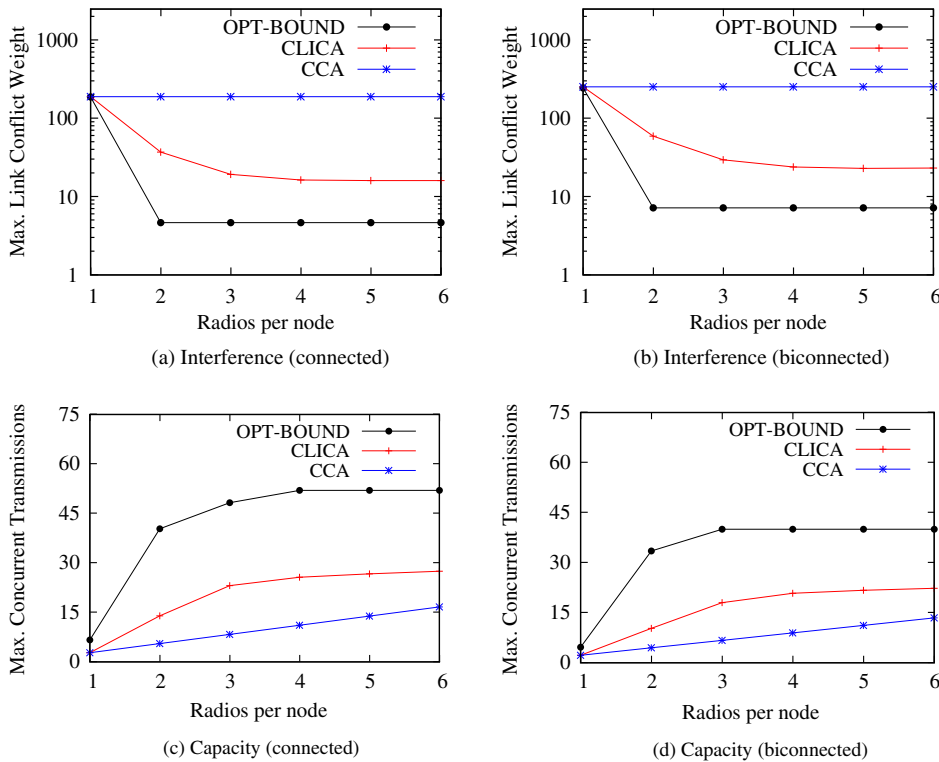


Fig. 6. Interference and capacity with CLICA and CCA algorithms with 12 channels and a varying number of radios per node and node densities.

imum number of concurrent transmissions) performance (Fig. 6c and d) shows a linear growth with increase in radios. Both these trends are expected given the way CCA assigns channels to radios. Recall that CCA does identical channel assignment at all nodes. As a result, the number of radios in the network assigned to a channel is the same as in the single channel case regardless of the number of radios available per node, which means potential interference remains identical to the single channel scenario. However, each additional radio at a node allows CCA to use an additional channel, hence there is a proportional increase in the capacity. Note that CCA capacity would be optimal when there are as many radios per node as the number of channels, an atypical scenario.⁴

CLICA performance is superior and markedly different from CCA. With CLICA, we note that the interference reaches a minimum value for a small number of radios per node regardless of the number of channels, but the absolute value for the minimum is smaller with more channels. The initial decrease in interference (Fig. 6a and b) is expected because of the added flexibility in choosing diverse channels with more radios. But after a point, the use of more radios does not reduce the interference as the number of channels becomes the bottleneck rather than the number of radios per node. Similar reasoning can be applied to the capacity performance with the CLICA algorithm (Fig. 6c and d) to explain the initial super-linear increase and marginal improvement thereafter.

Comparison of CLICA and bounds for interference and capacity shows that CLICA not only exhibits similar qualitative behavior as the bounds but is also comparable to the optimum bound. Note the logscale used for the interference plots. This is especially remarkable in the case of capacity because of the fact that we are using an approximation algorithm to determine the maximum number of concurrent transmissions (capacity) obtained with CLICA; this suggests that the actual performance gap with the optimum may be smaller.

Comparison at different densities, while not altering the qualitative behavior, validates intuition – higher densities lead to increased interference and reduced capacity. We have also studied scenarios where nodes have a different number of radio interfaces. Note that CLICA algorithm works for such scenarios without any modification. Comparing a homogeneous scenario (with the same number of radio interfaces at every node) to a heterogeneous scenario with an identical average number of radios per node, we observe that performance gets somewhat worse (higher interference and lower capacity) with the latter. This is, however, expected given that nodes with fewer radios per node constrain the channel choices for neighboring nodes to satisfy the connectivity constraint, which in turn pre-

vents the spreading of interference over a wide range of channels.

5.2. 802.11-based multi-radio mesh: single hop performance

We now use ns-2 simulations to evaluate the link layer performance of CLICA in a 802.11-based multi-radio mesh network, in terms of aggregate one-hop throughput and average delay (Fig. 7), focusing on connected scenarios. We present data for 3 and 12 non-interfering channels to study performance representative of 802.11b and 802.11a networks, respectively. These simulations use a commonly used 802.11 physical layer model in ns-2 that operates at a fixed data rate of 2Mbps. Even though higher data rates are available with 802.11b and 802.11a, the issue of physical layer data rate is orthogonal to our interests here – we are interested in relative performance improvements. The traffic model consists of unicast data with identical poisson packet arrivals between every pair of neighboring nodes in the network. Mean packet arrival rate is varied to obtain different offered loads, while keeping the packet size fixed (1KB). RTS/CTS mechanism is enabled. With CCA and CLICA, when a node can communicate with its neighbor via multiple radio interfaces tuned to different channels, we randomly stripe data across those interfaces (like one of the schemes in [6]). Each point in the plots is an average of five runs with different randomly generated node locations.

As seen from Fig. 7a and b, CLICA provides a significant improvement in throughput with multiple radios and channels compared to the single channel case – up to a factor of 3 with 3 channels and 2 radios, and a factor of 9 with 12 channels and 3 radios. For a comparable improvement, CCA needs as many radios as the number of channels. CCA throughput is much lower in relation to CLICA with fewer radios per node because the number of channels it can use is limited by the number of radios. The improvement in delay (Fig. 7c and d) over a single channel case is often dramatic with CLICA going up to a factor of 100 or more (not easily obvious due to scale of delay plots) for low to moderate traffic loads. This improvement factor is much more than the additional resources (radios and channels) used, which only go up to only a factor of 12 in these experiments. The primary reason behind this huge reduction in delay is due to the reduced interference (contention) and collisions, leading to smaller channel access delays (including back-offs) and retransmission delays in the 802.11 MAC and consequently smaller queueing delays.

5.3. 802.11-based multi-radio mesh: multihop performance

5.3.1. Diverse Channel Assignment and Inter-hop Interference

We again use ns-2 simulations to evaluate the effectiveness of CLICA algorithm for multihop communication on a path, where inter-hop interference is a key limiting factor. The use of multiple channels and radios can reduce such interference through the use of diverse channels at each hop, and thus allowing simultaneous reception and transmission on different radios at intermediate nodes. In these simulations, we use average end-to-end TCP throughput of

⁴ Exceptional cases are those with few channels available (e.g., 3 channels with 802.11b operating in 2.4 GHz band) when it may be possible to equip as many radios per node as the number of channels. However, the potential gains with multiple channels and multiple radios are limited in such cases, which justifies the emerging trend to have the backhaul tier of mesh networks operate in 5 GHz band where there are more channels available.

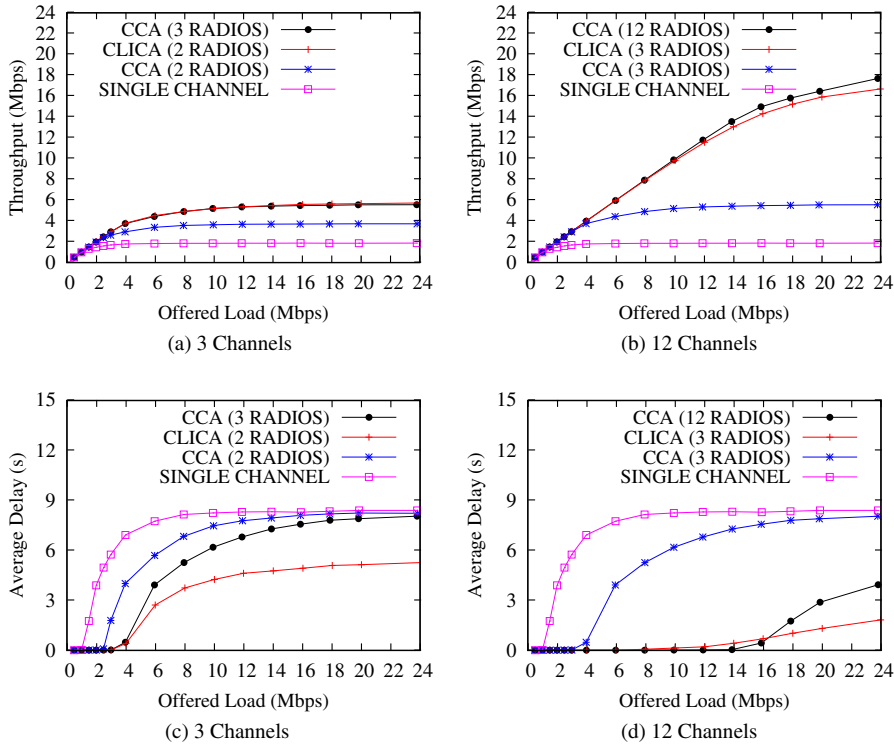


Fig. 7. Link layer performance (aggregate one-hop throughput and average delay) with CLICA and CCA algorithms in a 802.11 network relative to the single channel case.

a multihop path as the metric. For the application, we use 50 s one-way bulk transfer with FTP. We consider two different traffic patterns. For the Internet access pattern, we assume four randomly located Internet gateway nodes and simulate a data transfer to each non-gateway node from its nearest gateway node; nearness is determined by the shortest path length in hops, a good choice when all links have similar loss characteristics. For the peer-to-peer traffic pattern, we separately simulate a data transfer between 100 randomly chosen node pairs. In experiments with both types of traffic patterns, we disabled striping ability (i.e., distributing traffic to a neighboring node among multiple interfaces tuned to different channels) altogether in fairness to the single channel case and to isolate the benefit of using channel diverse paths. As before we use the 802.11 MAC.

Fig. 8a and b shows the results for the Internet access and peer-to-peer traffic patterns, with data averaged across samples with same path length. As expected, all cases have a similar performance for one-hop transfers. With longer paths and a greater number of channels and radios, CLICA provides larger throughput improvements over a single channel case (up to a factor of five). Note that CCA is not shown in these plots because the performance of CCA and a single channel case are indistinguishable in this scenario.

5.3.2. Effect of multiple flows

In real-world deployments, it is common to have multiple traffic flows (application sessions) active at the same

time. So we now consider the impact of multiple concurrent flows on network performance with various channel assignment algorithms. As in the previous single flow experiment, we use 50 s one-way bulk transfer with FTP. Similarly, we consider both Internet access and peer-to-peer traffic patterns. With Internet access pattern, each flow is from a gateway node to a nearby non-gateway node. Flows are setup between random node-pairs in the peer-to-peer traffic pattern.

Fig. 8c and d shows average TCP throughput performance with a varying number of flows for both cases, respectively. It is clear that CLICA consistently delivers a better performance than CCA, and both CLICA and CCA significantly outperform the single channel case. We can make two further observations from these results:

- (1) CLICA only provides a small improvement over CCA with fewer number of flows in the Internet access pattern (see Fig. 8c). This happens because:
 - Path lengths are relatively small with this traffic pattern (see Fig. 8a), which means that the opportunity for performance enhancement due to diverse channel assignment is limited.
 - CCA is fairly effective with a small number of flows because it can exploit the availability of multiple channels to communicate over each hop of a path and improve the performance of each of those flows via striping without hurting the throughput of other flows.

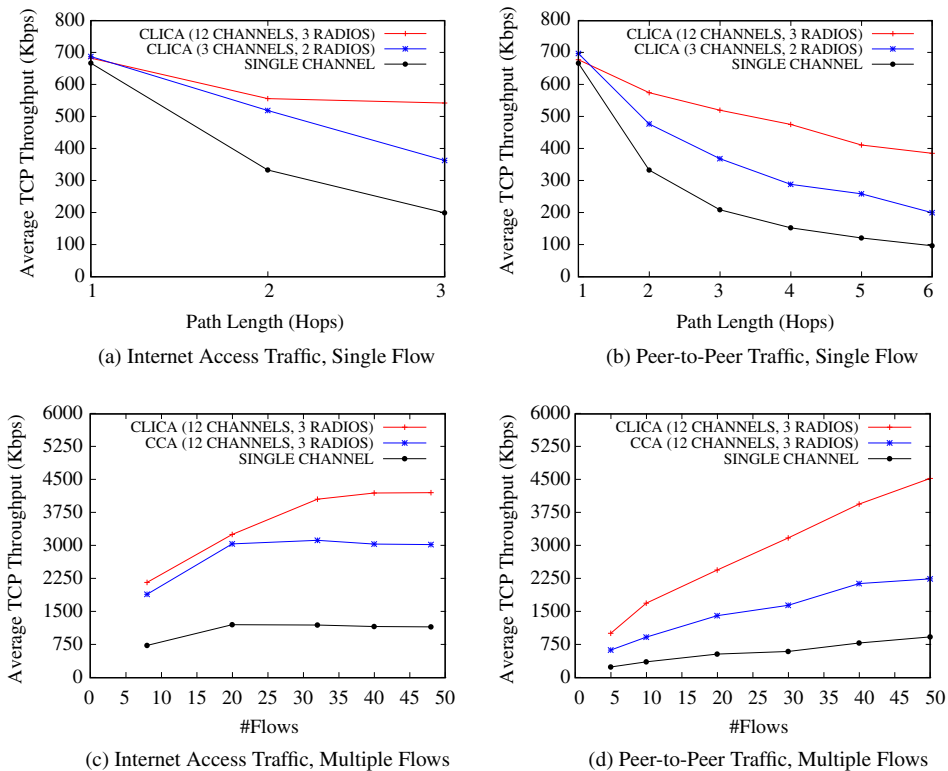


Fig. 8. Multihop TCP throughput performance for two different traffic patterns (Internet access and peer-to-peer) as a function of path length in the single flow case and varying number of flows in the multiple flow case.

- (2) For a given number of flows, the performance gain with CLICA relative to CCA is higher with the peer-to-peer traffic pattern because channel assignment in CLICA by default does not consider factors such as skewness in a traffic pattern and closeness of a node to a wired Internet gateway.

Both these observations imply ways to enhance performance of the CLICA algorithm. CLICA can be used to generate a base or default channel assignment that can be dynamically adapted based on the traffic pattern (e.g., switching to a CCA-like assignment temporarily in the lightly loaded scenarios). Additionally, the CLICA channel assignment algorithm can be improved by letting it consider traffic pattern and load information.

5.4. Summary

Our graph-based simulations in Section 5.1 show that CLICA can generate topologies with low interference with a small number of radios per node. The exact number of radios required, however, depends on the number of channels and node density. This issue requires further study. We also observe that interference and capacity trends with CLICA match that of corresponding optimum bounds. Furthermore, results for variable density and differing number of radios at nodes are along expected lines.

Reduced interference obtained via CLICA leads to a remarkable improvement in link layer network performance (in terms of throughput and delay) relative to the single channel case and CCA, as evaluated using ns-2 simulations in Section 5.2. Here it is noteworthy to mention that CLICA with a few radios has a similar or better performance compared to the case using as many radios per node as the number of channels.

Our evaluation of multihop path throughput with CLICA (in Section 5.3) shows that it is able to effectively alleviate inter-hop interference, thereby it can scale much better with increasing path length (or equivalently, network size) regardless of the traffic pattern.

The evaluation of the realistic case of multiple competing flows shows that CLICA is able to maintain its superior performance across the whole spectrum. Besides, this evaluation reveals ways to further optimize CLICA performance (and more generally, the performance of traffic-independent channel assignment algorithms) including adaptation to the traffic pattern (see Section 5.3).

We finally note that it was meaningful to do a quantitative comparison of CLICA only with CCA, since it is also traffic-independent and has identical assumptions and constraints. However, we would like to mention a qualitative comparison study from the literature [41] that includes CLICA and shows that, compared to other centralized channel assignment algorithms, CLICA also

has the property of preventing ripple effect during channel assignment due to its one-pass feature.

6. Conclusions and future work

In this paper, we have considered the channel assignment (radio-channel mapping) problem in multi-radio wireless mesh networks. Specifically, we have studied the channel assignment problem in the context of a traffic-independent framework. We have formulated the channel assignment as a topology control optimization problem where we have sought to minimize maximum interference in the network while preserving network connectivity, and showed it to be NP-complete. We also presented an ILP formulation of the problem for the purpose of deriving a lower bound for the optimum. We have then developed a greedy polynomial-time heuristic channel assignment algorithm called CLICA to find connected and low interference topologies. Combination of graph-based simulations and detailed ns-2 simulations demonstrate the effectiveness of the CLICA algorithm in exploiting channel diversity for reducing interference with a small number of radios per node, thus resulting in significant performance benefits in a 802.11-based multi-radio mesh network with single hop as well as multihop workloads. Our evaluation using graph-based simulations shows that network topologies obtained with CLICA-based channel assignment provide interference and capacity of similar behavior as the corresponding optimum bounds while being comparable.

Our future work will focus on further evaluation of the CLICA algorithm, specifically on theoretical performance characterization and real-world performance evaluation in a multi-radio mesh testbed. We will also be investigating the design of adaptive channel assignment techniques for large-scale and dynamic settings. Finally, investigation of channel assignment performance with diverse and mixed traffic patterns remains an issue for future work.

Appendix A

A.1. Physical model

In this model, a transmission from a node i to a node j is successful if the signal-to-noise ratio at j with respect to i 's transmission (SNR_{ij}) exceeds a threshold (SNR_{thresh}). SNR_{ij} is calculated by taking the ratio of received signal strength from i at j (SS_{ij}) to total noise at j ; the total noise includes aggregate received signal strength at j from all interfering transmitters as well as the ambient noise. For this model, the conflict graph is simply a complete graph since every pair of links in the network can interfere with each other; edge weights in the conflict graph quantify the amount of interference. One way to find those weights is as follows [22]. Let l_{ij} denote a node in the conflict graph corresponding to a link between node i and j in the actual network. The weight of a directed edge from l_{pq} to l_{ij} (denoted by w_{ij}^{pq}) is the ratio of SS_{pq} to maximum permissible noise at j to ensure that transmission from i to j is successful (i.e., $w_{ij}^{pq} = SS_{pq} / (SS_{ij} / SNR_{\text{thresh}})$, assuming no ambient noise). Note that for this model, the conflict graph is directed, even

if the corresponding network topology is undirected, because w_{ij}^{pq} need not be equal to w_{pq}^{ij} . Extending interference modeling using the physical model to multiple channels can be done as with the protocol model (see Section 3.1).

References

- [1] IEEE Std. 802.11. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.
- [2] I. Akyildiz, X. Wang, W. Wang, Wireless mesh networks: a survey, Elsevier Computer Networks 47 (4) (2005) 445–487.
- [3] R. Bruno, M. Conti, E. Gregori, Mesh networks: commodity multihop ad hoc networks, IEEE Communications 43 (3) (2005).
- [4] R. Wiggins, Myths and Realities of Wi-Fi Mesh Networking, Yankee Group Report, 2006.
- [5] P. Bahl, A. Adya, J. Padhye, A. Wolman, Reconsidering wireless systems with multiple radios, ACM SIGCOMM Computer Communications Review 34 (5) (2004) 39–46, Oct.
- [6] A. Adya et al., A multi-radio unification protocol for IEEE 802.11 wireless networks, in: Proceedings of the IEEE International Conference on Broadband Networks (BroadNets), 2004.
- [7] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: Proceedings of the ACM MobiCom, 2004.
- [8] A. Raniwala, K. Gopalan, T. Chiueh, Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks, ACM SIGMOBILE MC2R 8 (2) (2004).
- [9] M. Alicherry, R. Bhatia, L. Li, Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks, in: Proceedings of the ACM MobiCom, 2005.
- [10] M. Kodialam, T. Nandagopal, Characterizing the capacity region in multi-radio multi-channel wireless mesh networks, in: Proceedings of the ACM MobiCom, 2005.
- [11] J. Zhang, H. Wu, Q. Zhang, B. Li, Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks, in: Proceedings of the IEEE International Conference on Broadband Networks (BroadNets), 2005.
- [12] P. Kyasanur, N. Vaidya, Capacity of multi-channel wireless networks: impact of number of channels and interfaces, in: Proceedings of the ACM MobiCom, 2005.
- [13] A. Das, H. Alazemi, R. Vijayakumar, S. Roy, Optimization models for fixed channel assignment in wireless mesh networks with multiple radios, in: Proceedings of the IEEE SECON, 2005.
- [14] M.K. Marina, S.R. Das, A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks, in: Proceedings of the IEEE International Conference on Broadband Networks (BroadNets), 2005.
- [15] G. Xue, J. Tang, W. Zhang, Interference-aware topology control and QoS routing in multi-channel wireless mesh networks, in: Proceedings of the ACM MobiHoc, 2005.
- [16] K. Ramachandran et al., Interference-aware channel assignment in multi-radio wireless mesh networks, in: Proceedings of the IEEE Infocom, 2006.
- [17] A. Raniwala, T. Chiueh, Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network, in: Proceedings of the IEEE Infocom, 2005.
- [18] J. Zhu, S. Roy, 802.11 Mesh networks with two-radio access points, in: Proceedings of the IEEE ICC, 2005.
- [19] P. Kyasanur, N. Vaidya, Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks, ACM SIGMOBILE MC2R 10 (1) (2006) 31–43.
- [20] B. Ko et al., Distributed channel assignment in multi-radio 802.11 mesh networks, in: Proceedings of the IEEE WCNC, 2007.
- [21] P. Gupta, P.R. Kumar, The capacity of wireless networks, IEEE Transactions on Information Theory 46 (2) (2000) 388–404.
- [22] K. Jain, J. Padhye, V.N. Padmanabhan, L. Qiu, Impact of interference on multi-hop wireless network performance, in: Proceedings of the ACM MobiCom, 2003.
- [23] J. So, N. Vaidya, Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver, in: Proceedings of the ACM MobiHoc, 2004.
- [24] P. Bahl, R. Chandra, J. Dunagan, SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks, in: Proceedings of the ACM MobiCom, 2004.
- [25] I. Katzela, M. Naghshineh, Channel assignment schemes for cellular mobile telecommunication systems: a comprehensive survey, IEEE Personal Communications 3 (3) (1996).

- [26] A. Mishra, S. Banerjee, W. Arbaugh, Weighted coloring based channel assignment in WLANs, *ACM SIGMOBILE MC2R* 9 (3) (2005) 19–31.
- [27] S. Ramanathan, A unified framework and algorithm for channel assignment in wireless networks, *Wireless Networks* 5 (2) (1999) 81–94.
- [28] R. Chandra, P. Bahl, P. Bahl, MultiNet: connecting to multiple IEEE 802.11 networks using a single wireless card, in: *Proceedings of the IEEE Infocom*, 2004.
- [29] A. Ephremides, Ad hoc networks: not an ad hoc field anymore, *Wireless Communications and Mobile Computing* 2 (5) (2002).
- [30] M. Burkhart et al., Does topology control reduce interference? in: *Proceedings of the ACM MobiHoc*, 2004.
- [31] E. Lloyd et al., Algorithmic aspects of topology control problems for ad hoc networks, in: *Proceedings of the ACM MobiHoc*, 2002.
- [32] M.R. Garey, D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, 1979.
- [33] I. Holyer, The NP-completeness of edge-coloring, *SIAM Journal of Computing* 10 (1981).
- [34] R.L. Rardin, *Optimization in Operations Research*, Prentice-Hall, 1998.
- [35] R. Montemanni, D.H. Smith, S.M. Allen, Lower bounds for fixed spectrum frequency assignment, *Annals of Operations Research* 107 (2001).
- [36] S. Davis, R. Impagliazzo, Models of Greedy algorithms for graph problems, in: *Proceedings of the ACM-SIAM SODA*, 2004.
- [37] R. Ramaswami, K.K. Parhi, Distributed scheduling of broadcasts in radio network, in: *Proceedings of the IEEE Infocom*, 1989.
- [38] K. Fall, K. Varadhan (Eds.), *The ns manual*, 2002, <<http://www.isi.edu/nsnam/ns/ns-documentation.html>>.
- [39] H. Balakrishnan et al., The distance-2 matching problem and its relationship to the MAC-layer capacity of ad hoc wireless networks, *IEEE JSAC* 22 (6) (2004).
- [40] C. Bettstetter, On the minimum node degree and connectivity of a wireless multihop network, in: *Proceedings of the ACM MobiHoc*, 2002.
- [41] H. Skalli, S. Ghosh, S.K. Das, L. Lenzini, M. Conti, Channel assignment strategies for multiradio wireless mesh networks: issues and solutions, *IEEE Communications* 45 (11) (2007) 86–95.



Mahesh K. Marina is a faculty member in the School of Informatics at the University of Edinburgh, where he has been since November 2006. Before coming to Edinburgh, he spent around two years as a research staff member in the UCLA Computer Science Department. He received his Ph.D. in Computer Science from the State University of New York at Stony Brook in August 2004. Broadly, his research interests are in the areas of wireless and mobile networking, performance evaluation, distributed systems and

algorithms. More information about his current research activities can be obtained from <http://homepages.inf.ed.ac.uk/mmarina/>.



Samir R. Das is currently an Associate Professor in the Computer Science Department in the State University of New York at Stony Brook. He received his Ph.D. in Computer Science from Georgia Institute of Technology, Atlanta, in 1994. His research interests are in wireless networking and mobile computing, focusing on protocols, systems and performance evaluation. He received the U.S. National Science Foundation's CAREER award in 1998 and the best paper award in *ACM MobiSys* conference in 2007. He has been a speaker in the Distinguished Visitor program of the IEEE Computer Society during 2001–03. He co-chaired the technical program committee for the *ACM MobiHoc Symposium* in 2001 and *ACM MobiCom Conference* in 2004. He currently serves or has previously served on the editorial board of the *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Mobile Computing*, *ACM/Kluwer Wireless Networks Journal* and *Ad Hoc Networks* journal. More information about him and his research can be found at <http://www.cs.sunysb.edu/~samir>.



Anand Prabhu Subramanian is currently a Member of Technical Staff at Bell Labs, Alcatel-Lucent Inc. at Murray Hill, NJ. He received his Ph.D. in Computer Science from Stony Brook University, New York in 2009. Earlier, he received a B.E. degree in Computer Science and Engineering from Anna University, Chennai, India, in 2004 and an M.S. degree in Computer Science from Stony Brook University in 2007. His research interests include wireless networks, systems, and algorithms. He received the best paper award in *ACM MobiSys* Conference in 2007. His current research focuses on understanding the complex nature of wireless 3G/4G data networks.