The University of Sydney

A SURVEY ON CHANNEL ASSIGNMENT APPROACHES FOR MULTI-RADIO MULTI-CHANNEL WIRELESS MESH NETWORKS

TECHNICAL REPORT 613

WEISHENG SI AND SELVADURAI SELVAKENNEDY
SCHOOL OF INFORMATION TECHNOLOGIES
THE UNIVERSITY OF SYDNEY

AUGUST 2007

A Survey on Channel Assignment Approaches for Multi-Radio Multi-Channel Wireless Mesh Networks

WEISHENG SI and SELVADURAI SELVAKENNEDY
(weisheng, skennedy)@it.usyd.edu.au
School of Information Technologies, The University of Sydney, Australia

ABSTRACT

Channel Assignment (CA) has emerged as a new research area due to the rising commercial deployments of multi-radio multi-channel wireless mesh networks today. This paper presents an in-depth survey of the state-of-the-art CA approaches. Firstly, the key design issues for the CA approaches are identified. Secondly, a classification of the CA approaches that captures their essentials is proposed. Thirdly, these approaches are examined individually, with their advantages and limitations identified. As a summary, an overall comparison on them based on their basic properties is made. Finally, the future research directions for performing CA are suggested.

Additional Key Words and Phrases: interference, channel assignment, multiple radios, multiple channels, wireless mesh networks

1. INTRODUCTION

Wireless Mesh Networks (WMNs) have been adopted in many municipal/enterprise area network deployments according to the recent industry whitepapers [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005]. They are used in a variety of application scenarios such as the last-mile broadband Internet access, campus networks, mobile telephony backhaul networks, and public safety networks. With multiple hops and a mesh topology, the WMN architecture generally consists of three levels as described below.

- The top level comprises one or several gateways, which connect to both the WMN and the wired Internet and forward traffic between these two types of networks. This level can be absent if no Internet access is required and the WMN is only used for local communication.
- The intermediate level comprises numerous mesh routers, which are the vertices in the mesh topology and relay the traffic within the WMN. To enable wide and low-cost deployment, the link and physical layer protocols adopted by mesh routers are international standards such as the IEEE 802.11 protocol [IEEE 2003] or the IEEE 802.16 protocol [IEEE 2004] according to the current practice [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005].
- The bottom level comprises many WLANs or mobile phone cells depending on the usage of WMNs. Either a WLAN or a mobile phone cell generally consists of an Access Point (AP) and a certain number of wireless clients (e.g., laptops or mobile phones) that are the real producers and consumers of the wireless traffic.

Note that the gateways and the APs also have the full functionality of the mesh routers besides interfacing with the Internet and the wireless clients respectively, so they both play double roles. And in the study of WMNs, researchers mainly focus on the issues of mesh routers, leaving the Internet or the WLAN/cell issues to other areas of research. In this paper, we also refer to the mesh routers as "nodes" hereafter.

In the initial design of WMNs, the traditional wireless network paradigm is followed, where only one radio (i.e., wireless interface card[1]) is equipped at each node and all nodes share a single channel [Akyildiz *et al.* 2005, Bruno *et al.* 2005]. However, several research findings gradually revealed that the capacity per node in such solutions drops significantly with the increase of the network size. For example, [Gupta and Kumar 2000] demonstrated that when n identical randomly placed nodes with bandwidth W form a single-channel network, the throughput obtained per node is $\Theta(W/\sqrt{n \log n})$ asymptotically; ref. [Xu and Saadawi 2001] demonstrated that in a multi-hop network with all links running the same IEEE 802.11 protocol, the end-to-end performance suffers low throughput and unfairness problems. An intuitive interpretation of the above phenomenon is as follows: in a single-channel multi-hop network, interference can occur not only between nearby flows (known as inter-flow

---

[1] In this paper, we use "radio" and "interface" exchangeably, both of which refer to the "wireless interface card".

interference) but also between the nearby hops in a single flow (known as intra-flow interference), thus significantly degrading the network performance. To illustrate these two kinds of interference, Fig. 1 shows two concurrent flows in a multi-hop network, where not only flow 1 and flow 2 interfere with each other but also hop A→B and hop B→C in flow 1 interfere with each other.

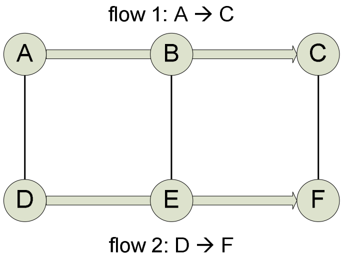


Fig. 1. Inter-flow and intra-flow interference

Due to the above observation, the single-radio single-channel architecture is not appealing. How about the single-radio multi-channel architecture then? It is also considered undesirable because of the following two reasons. First, with only a single radio, a node has to change its channel frequently with the dynamic network traffic so as to fully exploit the multi-channel advantage. Unfortunately, channel switching involves non-negligible delays. For instance, in the IEEE 802.11 hardware, this delay is usually in the order of several milliseconds [Raniwala and Chiueh 2005], which is quite significant compared with the transmission delays. Second, with the single radio assigned to different channels to reduce the interference, the nodes suffer low connectivity and even disconnectedness. Consequently, it is difficult to provide fault tolerance support or to synchronize neighboring nodes to the same channel if they want to communicate.

Therefore, almost all the current WMN deployments and proposals adopt the multi-radio multi-channel architecture, where each node is equipped with multiple radios and can use multiple non-overlapping channels. The WMNs with such architecture are generally called the multi-radio multi-channel WMNs, and for simplicity, we abbreviate them as M2WMNs hereafter. Fig. 2 illustrates the M2WMN architecture with an example network, where each node has two radios and can use four non-overlapping channels, with the channels reused spatially.

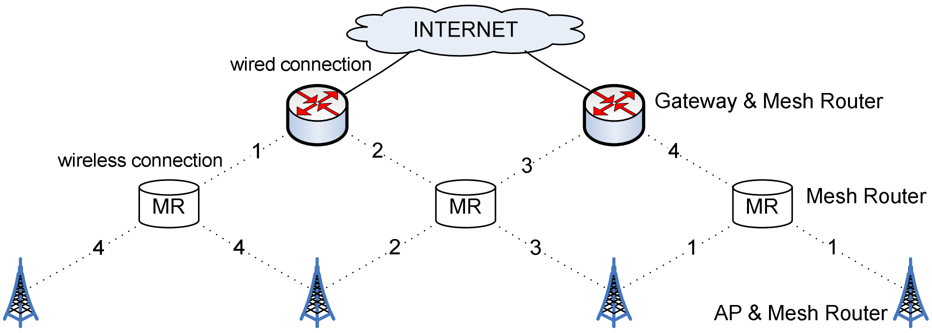


Fig. 2. The M2WMN architecture

Verified by the current commercial deployments [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005], this M2WMN architecture is practical due to the following two facts:

1) The cost of wireless interface cards has dropped rapidly with the proliferation of wireless networks, so the cost of multiple radios is no longer a prohibitive factor.
2) The current IEEE 802.11 and 802.16 standards both support multiple non-overlapping channels. For instance, there are twelve non-overlapping channels with 20MHz center frequency spacing in IEEE 802.11a [IEEE 1999] and three non-overlapping channels with 25MHz center frequency spacing in IEEE 802.11b [IEEE 2000]. Fig. 3 shows the three non-overlapping channels regulated in US; in Europe, the regulation is only slightly different [IEEE 2000]. For IEEE 802.16, it utilizes radio frequencies of both licensed and unlicensed bands from 2GHz to 66GHz[2] with a flexible channel bandwidth, so it can support significantly more non-overlapping channels than IEEE 802.11 [Ghosh *et al.* 2005]. Therefore, the availability of multiple non-overlapping channels is not an issue as far as the standardization is concerned.

---

[2] Not all are available due to the different regulations in different countries.

As an aside, [Paul *et al.* 2007] recently demonstrated through extensive real-world experiments that there actually exists significant interference between those standard non-overlapping channels in the current commodity IEEE 802.11 hardware. Nevertheless, Paul *et al.* also pointed out that this problem can be resolved by using better frequency filters in the hardware for multi-channel use.

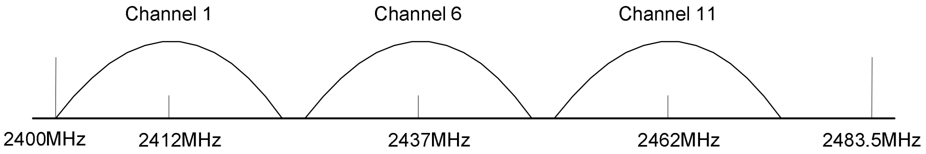


Fig. 3. North America Non-overlapping Channel Selection according to IEEE 802.11b

The advantages of M2WMNs are apparent. With each node equipped with multiple radios, multiple transmissions/receptions can happen concurrently, which multiplies the throughput. With multiple channels, neighboring links assigned to different channels can carry traffic free of interference, such that the link-layer delay can be dramatically reduced. As a result, when compared with the single-radio and single-channel solution, one of the proposed solutions [Raniwala and Chiueh 2005] shows that its multiple-radio and multiple-channel solution improves the network throughput up to a factor of seven, and an industry report [MeshDynamics 2006] claims that its multiple-radio and multiple-channel solution improves the network throughput by a factor of five.

However, the above advantages cannot be fully realized unless a number of issues related to M2WMNs are resolved. Generally, these issues include node deployment, channel assignment, link scheduling, and routing. Among them, the issue of channel assignment (CA), which aims to optimize the M2WMN performance by seeking a proper mapping between the available channels and the radios at every node, has received extensive attention. The CA issue is especially important due to the following reasons:

- It is intrinsic to M2WMNs and not present in other types of networks, thus launching a new research area.
- It is a must-solve problem for the M2WMN deployment.
- It is challenging because many formulations of the CA problem turn out to be NP-hard.

Therefore, we focus on the CA issue in this paper and present an in-depth survey of the state-of-the-art approaches addressing this issue. Our purpose is to provide insight into and foresee the future trends of this research area. The rest of this paper is organized as follows. To provide basis for examining these CA approaches, Section 2 specifies the M2WMN model and identifies the key design issues that confront the CA approaches. Section 3 proposes a classification of the CA approaches based on their basic characteristics. Section 4 describes the underlying idea and basic steps of all surveyed CA approaches individually, points out their advantages and limitations, and makes a comprehensive comparison on them. Section 5 presents the future research directions. Finally, Section 6 concludes this paper.

2. NETWORK MODEL AND DESIGN ISSUES

Before surveying the state-of-the-art CA approaches, we need to clarify the network model considered by them and identify the key design issues for them to address. Besides, this section also provides the necessary background for later discussions.

2.1 Network Model

All the approaches surveyed in this paper assume the following characteristics in their M2WMN models. Though each approach may possess its own distinct characteristics, the following ones are common.

1) The nodes in M2WMNs are not mobile.
2) At least some of the nodes in M2WMNs are equipped with multiple radios.
3) Multiple non-overlapping channels (free of inter-channel interference) are available to the network.

2.2 Key Design Issues

The characteristics in the above model actually distinguish M2WMNs from other types of wireless networks. With these characteristics, we identify the following key design issues for the CA approaches to address.

2.2.1 Interference. Interference is almost the foremost factor that degrades the wireless network performance, so the primary goal of CA is to minimize interference within the M2WMNs by utilizing the multiple radios and multiple channels. To address the interference issue, a model describing the interference effect needs to be assumed. Currently, there are two widely-adopted models: Protocol Model and Physical Model, both of which are initially defined in [Gupta and Kumar 2000]. The Protocol Model is simple, described as follows: (1) each radio has a transmission range and an interference range, with the former less than the latter; and (2) a transmission from radio X to radio Y is successful if Y is in the transmission range of X and not in the interference range of radios other than X that are currently transmitting. Compared with the Protocol Model, the Physical Model is close to reality but rather complex. It is described as follows: (1) a transmission is successful if the Signal to Interference and Noise Ratio (SINR) of the transmitter's signal at the receiver is larger than a threshold value; and (2) the interference and noise power at the receiver consists of the noises generated by other ongoing transmissions and the ambient noise in the network. Due to the complexity of the Physical Model, most of the CA approaches surveyed in this paper adopts the Protocol Model, only except [Mohsenian Rad and Wong 2006] that uses the Physical Model.

Besides the interference model, the following network constraints also need to be considered by the CA approaches in dealing with interference.

- The number of available channels: due to the technology constraints or government regulations, the M2WMNs actually cannot use as many non-overlapping channels as needed. Thus, interference cannot be completely eliminated from the M2WMNs.
- The number of radios at each node: it limits the number of non-overlapping channels that can be assigned to each node.
- The node deployment: it determines the geometric distances among the nodes. If the Protocol Model is assumed, the area within which the nodes interfere with each other is actually very large. This is because, to imitate the reality, the interference range is generally considered to be 2-3 times the transmission range. For instance, in both NS2 and QualNet (two popular simulators in use today), the default interference range is approximately twice the default transmission range [Xu *et al.* 2003].

So in addressing the interference issue, CA approaches need to consider at least the aforementioned factors: the interference model, the number of available channels, the number of radios at each node, and the node deployment. As to be described later, the joint consideration of these factors mostly results in NP-hard CA problems.

2.2.2 Connectivity. A terminology in graph theory, connectivity is defined on graphs, which are extensively used to model computer networks. To make our later discussion of connectivity issue unambiguous, here we describe and distinguish the following two graph concepts that are exploited by most of the existing CA approaches: *unit disk graph* and *network topology*. Note that, due to the terminological inconsistency of the literature, these approaches may not use the same terms when they refer to these two graphs.

The *unit disk graph* [Clark *et al.* 1990] is widely used to abstract the wireless networks under the Protocol Model. Specifically, a unit disk graph *G* (*V*, *E*) is defined as an undirected graph where (1) *V* is the set of nodes in the network and (2) $\forall v_1, v_2 \in V$, $(v_1, v_2) \in E$ if v_1 is in the transmission range of v_2 (also implying v_2 is in the transmission range of v_1). Note that, as suggested by the name of this concept, an identical transmission range for all nodes is assumed here. This assumption is reasonable in wireless networks because bidirectional communication is necessary in an unreliable wireless link: the sender generally requires acknowledgement from the receiver after transmitting the data.

While the *unit disk graph* is only concerned with the geometric distances among nodes, the *network topology* [Marina and Das 2005, Ramachandran *et al.* 2006] models whether any two nodes actually share a common channel as their communication link. Specifically, a network topology *T* (*V*, *E*) is defined as an undirected graph where:

- *V* represents the set of nodes and *E* represents the set of actual communication links in the network.
- $\forall v_1, v_2 \in V$, $(v_1, v_2) \in E$ if one of the v_1's radios and one of the v_2's radios share a common channel and they are within the transmission range of each other. Moreover, if the multiple radios at v_1 and v_2 share *n* common channels, there are *n* links $(v_1, v_2) \in E$, where *n* is a positive integer.

According to these two definitions, the unit disk graph is independent of the channel assignment, but the network topology, on the contrary, is determined only after the channel assignment is complete. This difference arises from the *communication constraint* that, in a multi-channel environment, two neighbor radios must share a common channel to communicate. Due to this constraint, altogether two kinds of discrepancies can occur between the unit disk graph and the network topology: (1) a link between two nodes in the unit disk graph is absent in the network topology if the radios on these two nodes are not assigned a common channel; (2) multiple links exist between two nodes in the network topology if multiple common channels are assigned to the radios on these two nodes. These two kinds of discrepancies are illustrated in

Fig. 4, where (1) there are three non-overlapping channels available (numbered 1, 2, and 3 respectively), (2) nodes A, C each have one radio and nodes B, D each have two radios, and (3) the channels assigned to these radios are shown inside the square brackets nearby the nodes.

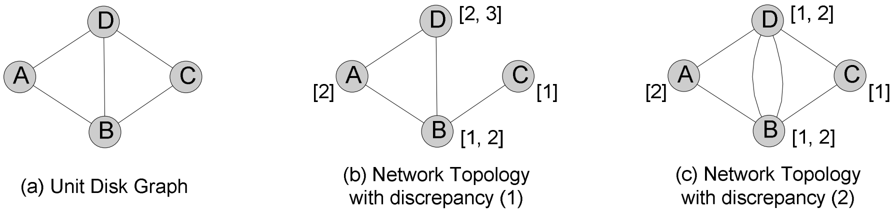


Fig. 4. Two kinds of discrepancies between unit disk graph and network topology

Though the unit disk graph and network topology differ in their structures, they are both important for the CA approaches. The former is usually used as the basis to perform channel assignment, since it is known initially and gives the distance relationship between the network nodes. And the latter is usually used to specify the connectivity requirement for channel assignment, since it actually carries the network traffic.

Therefore, in addressing the connectivity issue in M2WMNs, the CA approaches need to be aware that the CA decisions can actually change the network topology, which is a key difference between the single-channel and multi-channel networks. If the CA approaches ignore this difference, the network topology may be in danger of being disconnected, even though the interference is reduced by distributing channels to different radios. Thus, there is essentially a trade-off between interference and connectivity: the more radios are assigned to the same channel, the more connectivity is achieved, but the more interference is induced. As to be detailed later, all the approaches surveyed in this paper at least require that the network topology is connected after the channel assignment, while some approaches have stronger requirements. Consequently, how to minimize interference while satisfying the connectivity requirement is an important issue for the CA approaches to address.

2.2.3 Stability. The CA operation can cause two phenomena that undermine the network stability — *ripple effect* and *channel oscillation*, which need to be tackled properly by the CA approaches. The ripple effect, first described in [Raniwala and Chiueh 2005], also arises from the communication constraint that two radios in communication must share the same channel. The following example is given to exemplify the ripple effect. Assume interface *I2* originally at channel *x* wants to communicate with interface *I1* at channel *y*, thus *I2* switches to channel *y*. At same time, assume interface *I3* is currently communicating with *I2* using *x*, so *I3* has to switch to *y* to maintain the communication. And such channel change will continue to propagate, if another *I4* is currently communicating with *I3* using channel *x*. Another problem associated with the ripple effect is that, say in the above example, when *I2* switches to channel *y*, some packets may be lost in the communication between *I3* and *I2* on channel *x*, before *I3* switches to channel *y*.

A similar phenomenon to oscillation in routing, channel oscillation means that the channel assignment does not converge and changes back and forth among several choices. This phenomenon usually happens when the CA is based on a dynamic metric. For example, when two nodes discover that a channel is under-utilized according to such a dynamic metric, they may simultaneously switch to this channel and both begin transmission on it, and then switch back because this channel is now overloaded, as indicated by this dynamic metric. Since channel switching involves significant overhead such as switching delay and traffic interruption, frequent channel switching resulted from oscillation will severely impair the network performance.

2.2.4 Throughput/Latency. Throughput and latency are two most important measures for network performance. Having a deterministic relationship, they are generally addressed together. To obtain optimal throughput/latency in M2WMNs, the CA approaches need to consider the following two basic strategies. First, reducing interference is almost the most effective method in achieving the optimality, and it is better to make this method adaptive to the dynamic network traffic. Second, links should be treated differently when assigning channels, since different links in M2WMNs impact throughput/latency at different extents. For example, the backbone links carry much more network traffic than the stub links, so they should be given more bandwidth either by assigning more number of concurrent channels or by assigning less interfered channels. In this sense, an M2WMN is analogous to a wired enterprise/campus network consisting of a hierarchy of Ethernet switches, where the ports on an upper-level switch usually have much more bandwidth than the ports on a lower-level switch.

2.2.5 Routing. As shown in Subsection 2.2.2, the network topology of an M2WMN — a basic factor for making the routing decisions — can be changed by the CA decisions. Thus, routing is dependent on CA. On the other hand, routing can change the traffic load distribution in the network, which is a primary factor considered by CA to reduce the interference dynamically. So CA is also dependent on routing. With CA and routing mutually dependent, how to combine these two mechanisms to obtain optimal network performance is a very challenging issue.

2.2.6 Fault Tolerance. Though the nodes in M2WMNs are stationary, they can fail because of software or hardware problems. Moreover, the wireless links can also fail due to unexpected scenarios such as external interference or temporary obstacles. So it is necessary for a CA approach to support fault tolerance such that the network could operate in a self-healing fashion. Furthermore, though the multiple radios and multiple channels in M2WMNs provide abundant choices for recovering from faults, a selection among these choices has to be made to obtain the optimal results. Therefore, the fault tolerance issue is not an easy task for the CA approaches to address.

2.2.7 Fairness. To obtain the optimal overall network performance, fairness among the nodes is sometimes sacrificed in designing a CA approach. But in many scenarios such as the mobile phone backhaul networks, fairness is a necessary property of the network services. A basic fairness criterion for CA is the capability to avoid that the traffic of some nodes only has access to crowded channels shared by many links, while the traffic of other nodes has access to only partly-occupied channels shared by a few links. Though guaranteeing fairness to certain level introduces additional difficulty to the CA design, it is worth achieving in practice.

2.3 Summary to the Key Design Issues

All the CA approaches discussed in this paper aim to solve some of the above issues. Though these issues are listed as separate entries, they are inevitably related to each other. Some of the issues have mutually-beneficial relationships such as ‘less interference results in more throughput’ and ‘richer connectivity offers better fault tolerance’. However, there also exist trade-off relationships among these issues. For example, reducing interference gives rise to less connectivity, and supporting fairness sacrifices overall throughput. Generally, if both issues with a trade-off relationship are to be addressed, the balance needs to be made according to the goals or requirements of concrete CA approaches.

3. CLASSIFICATION OF CA APPROACHES

Despite a nascent research area starting around 2004, CA for M2WMNs has seen a decent number of approaches up to now. While possessing considerable similarity, these approaches also exhibit significant diversity. To deepen the understanding of them, we present a classification on these approaches in this section (see Fig. 5). Note that there can be different classifications according to different criteria, and our taxonomy is what we believe the best for capturing the essentials of these CA approaches.

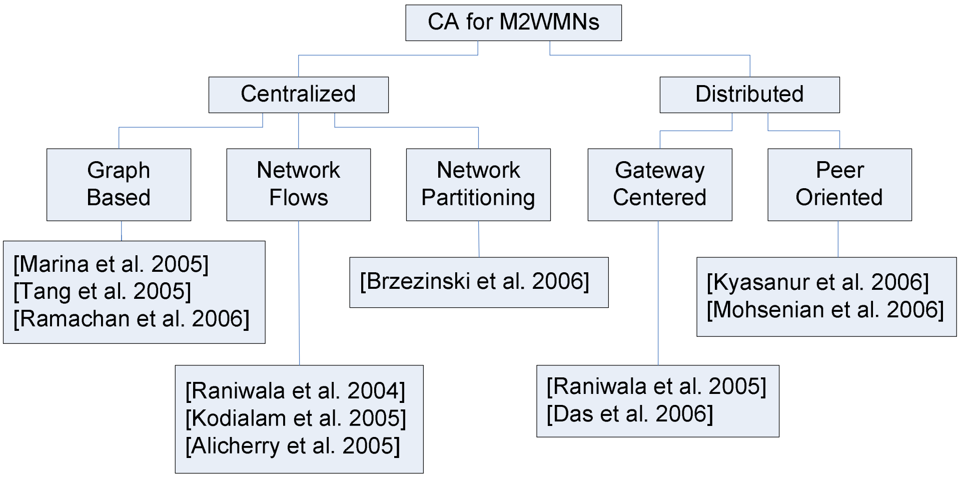


Fig. 5. Classification of CA approaches for M2WMNs

In general, all the CA approaches are classified into two categories: centralized or distributed. For the centralized approaches, a central control is assumed and it has the complete information of the network needed by the approaches. Thus, a CA problem can be formulated into a standalone problem, which is amenable for solving optimally or sub-optimally at a central point. After the result of CA is calculated, it is distributed to the nodes to accomplish. Altogether, there are three types of problems being formulated: the graph-based problem, the network flows problem, and the network partitioning problem. Accordingly, the centralized approaches are further classified into three categories which exploit the above problem formulations respectively. The details of the problem formulations and the algorithms proposed to solve them will be detailed in the next section.

For the distributed approaches, no central control is assumed and each node runs its own copy of the protocol to assign the channels. They are further classified into two categories according to the traffic pattern being considered: gateway-centered CA approaches and peer-oriented CA approaches. The former assumes that the main network traffic is to or from the gateways, so the CA can exploit the heuristic that the near-gateway links should be given relatively high bandwidth. The latter assumes that the network traffic can occur between any pair of nodes with no fixed pattern, so the CA approaches have to be as general as possible to accommodate various kinds of network traffic in a peer-to-peer fashion. The details of these approaches will also be discussed in the next section.

4. DESCRIPTION AND COMPARISON OF CA APPROACHES

In this section, we provide in-depth description and comparison of CA approaches based on our classification presented in Section 3. The centralized approaches are examined first, followed by the distributed approaches. For every approach, we extract its basic ideas, summarize its basic operation steps, and identify its advantages and limitations. In the end, a comprehensive table that compares and contrasts all the CA approaches based on their basic properties is presented.

4.1 Centralized CA Approaches

Assuming the availability of complete information at a central point, the centralized CA approaches are introduced before their distributed counterparts. From the literature, there are only centralized approaches but no distributed ones proposed in the year 2004. Next, we first provide the common background for all centralized approaches, and then explore them individually according to their categories.

As mentioned in Section 3, the centralized approaches are classified into three categories according to their problem formulations: the graph-based approaches, the network flows approaches, and the network partitioning approaches. To solve a problem centrally, only a standalone algorithm and no communication protocols are required. Thus, our classification on the centralized approaches is actually a classification on the standalone algorithms. We observe that inputs and outputs are two basic characterizing aspects within each category of algorithms. That is, though the algorithms in each category address the same type of problem, they can still be different in the inputs considered and the outputs produced. For the outputs, since all of them are essentially channel assignment results, they only vary in their objectives accomplished. For example, some outputs aim to maximize the overall throughput, while others aim to minimize the overall interference. For the inputs, however, there are many parameters from the networks to consider. Specifically, all the approaches discussed in this section consider some, if not all, of the following parameters as their inputs.

- Node deployment: the geometric position of each node in the network.
- Numbers of radios and channels: the number of radios at each node and the number of non-overlapping channels available at each radio.
- Interference model: all the centralized approaches surveyed in this paper adopt the Protocol Model, with some of them claiming that their schemes can also be generalized to support the Physical Model.
- Traffic profile: the bandwidth of each link and the end-to-end traffic rate of each flow.
- Connectivity constraint: the level of network connectivity to be achieved. All the approaches at least require the network topology to be connected after the channel assignment, while some approaches have stronger requirements such as k-connectedness in [Tang *et al.* 2005].
- Fairness constraint: specifying certain fairness requirement to be satisfied. For example, [Alicherry *et al.* 2005] requires that λ times the traffic rate of each source node can be routed over the network after the channel assignment, where λ is a constant for all the nodes.

It is worth noting that, despite different problem formulations, different sets of inputs, and different objectives of outputs, most centralized approaches (only except [Brzezinski *et al.* 2006]) surveyed in this paper state or formally prove [Marina and Das 2005, Ramachandran *et al.* 2006, Raniwala *et al.* 2004] that their formulated problems are NP-hard. Consequently, they resort to heuristic or approximation algorithms to get suboptimal results such that the computation complexity is polynomial. Thus, we consider the heuristic or approximation methods as a third characterizing aspect within each category of approaches.

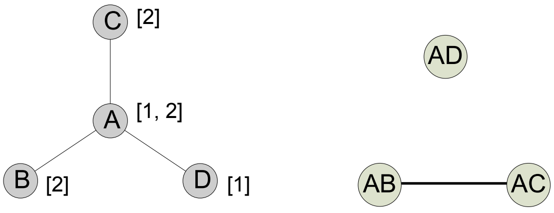
Therefore, when these three categories of approaches are examined in the following subsections, special attention is paid to the three aspects (inputs, outputs, and heuristic or approximate methods) in contrasting them within each category. The choice of these aspects is actually natural since a well-accepted definition of 'algorithm' is that 'a sequence of computational steps to transform inputs into outputs' [Cormen 2001], which exactly encompasses these three aspects.

4.1.1 Graph-based Approaches. In graph-based approaches, an M2WMN is modeled by a graph with a vertex set and an edge set, and the CA problem is formulated into the problem of assigning channels to vertices or edges in this graph. Three graph concepts are generally exploited by this category of approaches: *unit disk graph*, *network topology*, and *conflict graph*. With the first two concepts already described in Section 2, we next detail the concept of *conflict graph* G_C (V_C, E_C), which is first introduced within the M2WMN context in [Jain *et al.* 2003].

To facilitate the understanding of this concept, its basic ideas are given as follows: (1) it is derived from the network topology T (V, E) and models whether two links in E of T interfere with each other; (2) any link in E of T is represented by a vertex in V_C; and (3) if two links interferes, an edge connecting the two corresponding vertices in V_C is included in E_C to show this conflict. With the above said, the definition of *conflict graph* G_C (V_C, E_C) as an undirected graph under the Protocol Model is given as follows:

- $\forall\,(v_1, v_2) \in E$ of T, there is a vertex $v_{1,2} \in V_C$.
- $\forall\,(v_p, v_q),\ (v_s, v_t) \in E$ of T (i.e., $v_{p,q}, v_{s,t} \in V_C$), $(v_{p,q}, v_{s,t}) \in E_C$ if (1) v_p or v_q is in the interference range of v_s or v_t, or vice versa and (2) (v_p, v_q), (v_s, v_t) use the same channel.

According to the second condition of this definition, the bidirectional communication is assumed here: if either direction of the communication is interfered, the two links are considered to conflict with each other. To simplify our description, when this second condition is referred to hereafter, we simply say 'two links interfere with each other'. An example of network topology and its derived conflict graph is illustrated in Fig. 6, where (1) two channels are available, numbered 1 and 2 respectively, (2) node A has two radios and nodes B, C, D have one radio, and (3) the channels assigned to these radios are shown inside the square brackets nearby the nodes. As shown in Fig. 6 (a), there are three links in the network topology, so three corresponding vertices exist in the conflict graph depicted in Fig. 6 (b). Since only links AB and AC (both using channel 2) interfere with each other, there is one edge connecting the two corresponding vertices in the conflict graph.



(a) Network Topology (b) Conflict Graph

Fig. 6. An example of network topology and its derived conflict graph

As an aside, in most of the literature, the elements in V and V_C are called nodes and vertices respectively, and the elements in E and E_C are called links and edges respectively. For clarity, this naming convention is followed throughout this paper. Next, three approaches belonging to this category are examined: (1) connected low interference channel assignment [Marina and Das 2005], (2) minimum interference survivable topology control [Tang *et al.* 2005], and (3) breadth first search channel assignment [Ramachandran *et al.* 2006], all of which base their algorithms on the above graph concepts.

(1) Connected Low Interference Channel Assignment (CLICA)

To motivate its problem formulation, [Marina and Das 2005] proposes a novel notion called *base channel assignment*. This notion can be summarized by the following three characteristics: (1) it is independent of the traffic pattern and intended for use during the network deployment or maintenance stage; (2) it results in a network topology that any link in the unit disk graph is preserved; (3) the non-overlapping channels are distributed among different links such that the interference is reduced. The rationale for proposing this notion includes two observations made in [Marina and Das 2005]: (1) it is impractical to assume the traffic profile can be known *a priori* such that the CA algorithms can be optimized according to the traffic profile; (2) the availability of the *base channel assignment* simplifies the coordination among the nodes, thus enabling the next-stage dynamic and efficient channel assignment that can adapt to the network traffic.

With the *base channel assignment* problem as its sole focus, [Marina and Das 2005] formulates it into the problem of assigning channels to the links in the unit disk graph (called the *connectivity graph* in [Marina and Das 2005]). Specifically, the problem formulation in [Marina and Das 2005] has the following two output objectives: (1) any link in the unit disk graph possesses two adjacent radios that are assigned a common channel, which is referred to as the 'connectivity preservation' hereafter; (2) the interference over the whole network is minimized in the sense of minimizing the maximum *link conflict weight* among all links in the resultant *network topology*. Note that the *link conflict weight* for a link is defined as the sum of the number of edges incident to the vertex representing this link in the *conflict graph*.

After proving that the above formulated problem is NP-hard, [Marina and Das 2005] proposes a polynomial time heuristic algorithm named Connected Low Interference Channel Assignment (CLICA) to solve it. The basic steps of CLICA are summarized as follows:

1. Randomly assign a node v the highest priority, then assign other nodes priorities decreasingly in the order obtained by depth-first searching the unit disk graph starting from the node v. That is, a node v_1 is assigned a higher priority than a node v_2, if v_1 is traversed earlier than v_2 in the depth-first search.
2. While traversing the nodes in the decreasing order of their priorities obtained above, assign channels to the incident links of these nodes.
3. When a node is traversed in the above step, channels are assigned to all links incident to this node in the unit disk graph. The operation of assigning a channel to link includes assigning this channel to both a radio at this node and a radio at the neighbor node. Then, the priorities of unvisited nodes are adjusted according to their *degree of flexibility*, which is the number of channels that a node can choose from without breaking the connectivity preservation. Essentially, the nodes with less degree of flexibility will have their priorities increased so that they are visited earlier in the later steps.
4. When picking a channel in the above step, a node v_1 picks a channel for its incident link (v_1, v_2) in a greedy manner: a locally optimal choice is made by selecting the channel that minimizes the maximum link conflict weight among all links that can interfere with link (v_1, v_2). After a channel is assigned to a link, the conflict graph is updated to reflect the new *link conflict weight*s.
5. After the above traversal, there may still exist some nodes with unassigned radios because they have more radios than their number of incident links. These radios are assigned channels using the following method. Each unassigned radio at a node is paired with another unassigned radio at a one-hop neighbor node, and a channel is assigned to these pair of radios using the same greedy method as described in Step 4. If such a pairing is not possible, an unassigned radio is paired with an already-assigned radio at a one-hop neighbor node and is assigned the channel used by this already-assigned radio.

The proposed CLICA algorithm mainly has two advantages: (1) the paper proves that it achieves the connectivity preservation objective specified in the problem formulation; (2) it is a simple algorithm with a locally greedy heuristic, such that it can be implemented efficiently. On the other hand, the CLICA algorithm mainly has two limitations: (1) by using the unit disk graph to assign channels, the number of radios at each node is difficult to model, so an additional step in the end of the algorithm is needed to deal with those unassigned radios; (2) fairness may be sacrificed because it picks a channel for a link in a locally optimal fashion.

(2) minimum INterference Survivable Topology Control (INSTC)

Ref. [Tang *et al.* 2005] is similar to CLICA in that it also formulates the CA problem into the problem of assigning channels to links in the unit disk graph. To introduce a metric for minimizing the interference, [Tang *et al.* 2005] defines a concept called *Link Co-channel Interference (LCI)* for a link in the *network topology*. Basically, the LCI of a link means the number of links that interfere with this link in the network topology. Thus, the LCI concept is equivalent to the concept of *link conflict weight* defined by leveraging the conflict graph in CLICA. This equivalence exemplifies a worth-noting phenomenon that two parallel works coincidentally define an identical concept using different ways, which sometimes happens in the research community.

With the LCI metric, [Tang *et al.* 2005] formulates the CA problem into the 'minimum INterference Survivable Topology Control (INSTC)' problem, which is detailed as follows. Given the unit disk graph G, the number of radios Q at each node, the total number of channels C, and an integer k, seek a channel assignment such that the resultant network topology T satisfies the following two objectives: (1) T is k-connected so that the network is survivable to node/link failures; (2) the interference over the whole network is minimized in the sense that the maximum LPI for all links in T is minimized. Note that (1) a graph is said to be k-connected, if it remains connected after any less than k vertices are removed from this graph and (2) for this problem formulation to have a solution, the given G must be k-connected.

It can be seen that the second objective regarding interference in this formulation is the same as the second objective in CLICA's problem formulation, since the LPI concept is equivalent to the *link conflict weight* concept. The differences between these two problem formulations are that (1) INSTC requires all nodes to have identical number of radios Q, while CLICA allows variable number of radios at each node and (2) INSTC imposes a stronger connectivity requirement by k-connectedness than CLICA, which only requires that any link in the unit disk graph is preserved.

Due to the NP-hardness of the INSTC problem, [Tang *et al.* 2005] proposes a heuristic polynomial time algorithm (named the INSTC algorithm) to solve it. The basic idea of the INSTC algorithm is to assign the channels by traversing the links in a k-connected subgraph of the given unit disk graph in a predetermined order. To determine this order, [Tang *et al.* 2005] introduces another concept called *Link Potential Interference (LPI)* for a link in the unit disk graph. Simply put, the LPI for a link is defined to be the number of links that interfere with this link in the unit disk graph by only considering the interference range. Then, the links in the unit disk graph is sorted in a non-increasing order of their LPIs, thus obtaining the predetermined order used by the link traversal. As a note, LPI is used instead of LCI to determine the traversal order because the network topology is not known initially and hence the LCI is not available at the beginning of the algorithm.

With the above background, the basic steps of the INSTC algorithm can be summarized as follows:

1. Determine the link traversal order as mentioned above, then find a subgraph of G, denoted by G', such that G' is k-connected and includes all the links that have the minimum LPI.
2. Traverse the links in G' in the non-decreasing order of their LPIs and assign channels to them.
3. When picking a channel for a link e in the above step, a locally optimal choice is made — picking the channel that has been used least times among all links that interfere with e.
4. After the traversal of G', assign channels to the remaining unassigned radios in G. The method of picking a channel for a radio here is to choose the channel that has been used the least times among the already-assigned channels at the one-hop neighbor nodes of this radio.

As seen above, the behaviors of the INSTC and CLICA algorithms are similar in that they both pick a channel for a link in a locally optimal manner, and are different in that INSTC uses a predetermined order to traverse the links, while CLICA dynamically adjusts its traversal order according to the degree of flexibility obtained from the constantly updated conflict graph.

Besides the above channel assignment algorithm, [Tang *et al.* 2005] also proposes a routing algorithm based on the network topology generated by the CA algorithm. Though the CA algorithm does not consider the traffic load information, the routing algorithm is capable of adapting to the change of traffic load and also provides QoS support. Limited by the scope of this paper, the details of this routing algorithm are omitted here.

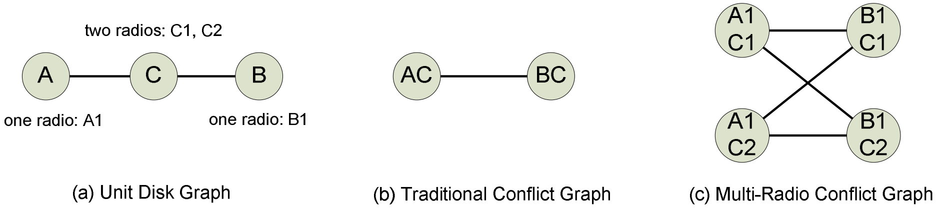
The main advantage of the INSTC algorithm is that it is proved to be a polynomial time algorithm that achieves k-connectedness, thus the algorithm is efficient and the obtained network topology is robust. Due to its close similarity with CLICA as revealed above, its limitations are the same as those of CLICA.

(3) Breadth First Search Channel Assignment (BFS-CA)

As described above, both CLICA and INSTC have the difficulty in modeling the number of radios at each node by using the unit disk graph. Ref. [Ramachandran *et al.* 2006] overcomes this difficulty by introducing the concept of Multi-radio Conflict Graph (MCG), which explicitly includes the number of radios at each node by extending the traditional conflict graph concept. The MCG differs from the traditional concept in two ways. First, it considers links between *radios* as vertices instead of considering links between *nodes* as vertices. Second, it models the interference between links based on the *unit disk graph* instead of the *network topology*. The reason for this is that [Ramachandran *et al.* 2006] works by assigning channels to the vertices of MCG, so the MCG needs to be available at the beginning of the algorithm. The definition of MCG (V_M , E_M) under the Protocol Model is given below.

- For each pair of radios that are adjacent to a link in the *unit disk graph*, there is a vertex representing this link in V_M.
- If two links interfere with each other, there is an edge in E_M connecting the two vertices that represent these two links to indicate this conflict.

To understand this concept, Fig. 7 gives a comparison of the traditional conflict graph and the MCG. Since the conflict graph is actually based on the network topology, we assume here that radios A1, C1, B1 use an identical channel and radio C2 uses a different channel in (a). Thus, only links AC and BC interfere with each other in (b). The MCG for (a) independent of the channel assignment is given in (c), where each pair of radios within transmission range constitutes a vertex, thus there are altogether four vertices representing links and four edges representing conflicts in this MCG. Because radio A1 can only be assigned one channel, links A1C1 and A1C2 can not exist simultaneously, so they do not conflict with each other, and the same applies to links B1C1 and B1C2.



(a) Unit Disk Graph (b) Traditional Conflict Graph (c) Multi-Radio Conflict Graph

Fig. 7. A comparison of the traditional conflict graph and the MCG

The problem formulation in [Ramachandran *et al.* 2006] makes the following assumptions on the M2WMN environment: (1) there is a gateway in the M2WMN and the major network traffic is to or from the Internet via this gateway; (2) there is a Channel Assignment Server (CAS) running on the gateway, and its task is to collect information from all nodes in the network, calculate the CA results, and distribute the results to all nodes; (3) each node at least has a default radio operating on a common default channel, which can carry both control and data packets; (4) there exist co-located external wireless networks that interfere with this M2WMN, since the M2WMN today mostly use the unlicensed RF bands. With these assumptions, [Ramachandran *et al.* 2006] formulates the CA problem into the problem of assigning channels to the vertices in MCG such that the following two objectives are satisfied: (1) minimizing the interference among the mesh routers and (2) minimizing the interference between the M2WMN and the co-located external wireless networks.

After stating that the formulated problem above is NP-hard, a polynomial time heuristic algorithm named Breadth First Search Channel Assignment (BFS-CA) is proposed to solve it. The main idea of BFS-CA is to traverse the MCG in BFS order and pick channels for vertices in MCG greedily according to the channel ranking based on the interference level. Its basic steps are summarized as follows.

1. Each node measures the external interference level of each channel it uses in terms of the number of external radios sharing this channel and the channel utilization by these external radios, and then sends this information to the CAS.
2. After receiving the above information from all nodes, CAS ranks all the channels in a decreasing order according to their interference levels reported by all nodes — the channels with less overall interference levels are ranked higher.
3. With the links adjacent to the gateway as the starting points, the algorithm traverses vertices (i.e., links between two radios) in MCG in BFS order, using the distance of a link from the gateway as the guideline. The distance of a link is measured by averaging the hops of this link's two adjacent radios from the gateway. Note that the purpose of BFS traversal here is to give the near-gateway links high priority to be assigned less-interfered channels, thus offering them more bandwidth.
4. While visiting a vertex in the above order, the vertex is assigned the currently highest ranked channel that is not assigned to its adjacent vertices in MCG. The rationale for this heuristic of assigning channels is as follows: (1) considering the MCG constraint is to reduce the interference among the mesh nodes, and (2) considering the channel rankings of the external interference level is to reduce the interference between the mesh and the external wireless networks. If such a channel is not available, then randomly assign a channel to this vertex.

Two additional notes are worth making about the BFS-CA algorithm. First, after a vertex in MCG is assigned a channel, all unassigned vertices in MCG that contain either of the two radios from the just-assigned vertex are removed. This is to ensure that only one channel is assigned to each radio in the mesh network. For example, after assigning a channel to vertex A1C1 in Fig. 7 (c), all vertices containing either A1 or C1 (i.e., A1C2 and B1C1) are removed, thus preventing radios A1 and C1 from being assigned a channel again. Second, the BFS-CA algorithm is executed periodically at the CAS, with the period configured to be 10 minutes as mentioned in its paper, so as to adapt to the dynamic external interference.

The advantages of the BFS-CA algorithm mainly include the following: (1) a novel concept of Multi-radio Conflict Graph is proposed, such that it is straightforward to consider the number of radios at each node in designing a CA algorithm; (2) it is claimed in [Ramachandran *et al.* 2006] that BFS-CA is the first algorithm considering the external interference, which is a practical problem in the current M2WMN deployments; (3) the practicality of this algorithm is demonstrated by a prototype implementation in a multi-radio IEEE 802.11b testbed. On the other hand, the BFS-CA algorithm has the following two limitations: (1) as seen from step 4 of the BFS-CA algorithm, its heuristic to reduce both internal and external interference by combining the channel ranking and the MCG constraint is intuitive, providing no known bound for the worst-case performance; (2) it is only suitable for the M2WMNs where a gateway acts as the central point of the network traffic.

(4) Summary to the graph-based approaches

As seen above, the three graph-based algorithms all use certain kind of graph to model M2WMNs and assign channels with a greedy heuristic. To clarify the relationship among these three algorithms, we compare and contrast them based on the inputs, outputs, and heuristic methods in Table I.

Table I. Comparison and contrast of the three graph-based algorithms

	Inputs	Outputs	Heuristic Methods
CLI CA	• The unit disk graph • Number of radios at each node and total number of channels • Interference reflected by the conflict graph	Channel assignment such that any link in the unit disk graph is preserved and the maximum link conflict weight among all links is minimized.	• Assign channels to the links in unit disk graph • DFS traversal, considering the degree of flexibility to adjust the order • Pick channels greedily according to *link conflict weight*
INS TC	• The unit disk graph • Fixed number of radios Q at a node and total number of channels C • Interference information reflected by LCI and LPI	Channel assignment such that k-connectedness is achieved and the maximum LCI among all links is minimized.	• Assign channels to the links in unit disk graph • A k-connected subgraph traversal, using a predetermined order based on LPI • Pick channels greedily according to LPI
BFS -CA	• The unit disk graph • Location of the gateway • Number of radios at each node and total number of channels • Internal interference reflected by the MCG • External interference measured by mesh routers	Channel assignment such that both internal interference among mesh routers and external interference between mesh and co-located wireless networks are reduced.	• Assign channels to vertices in MCG • BFS traversal, using the distance from the gateway as the guideline • Pick channels greedily according to the MCG and the external interference level

Besides the above comparison table, we also point out the advantages and limitations of the graph-based approaches as a category below.

- Advantage: graph models are intuitive and convenient for designing the CA algorithms.
- Limitation: solely modeling the network by vertices and edges, they have the difficulty to consider the traffic load information, an important factor for a CA algorithm to further improve the network performance.

4.1.2 Network Flows Approaches. In network flows approaches, an M2WMN is modeled by a flow network, thus overcoming the aforementioned limitation of the graph-based approaches. Network flows is a discipline originated from the 19th century that has applications in many fields. Here we only describe two basic definitions in this discipline: *flow network* and *flow*, which are essential for understanding the later discussions. For the details of network flows theory, two classical books [Ahuja *et al.* 1993] and [Cormen 2001] are recommended.

Specifically, a *flow network* is a directed graph G (V, E) in which each link $(u,v)\in E$ has a capacity $c(u,v)\geq 0$. If $(u,v)\notin E$, then $c(u,v)=0$. There are two special subsets of V: source set and destination set, denoted by V_s and V_d respectively. A *flow* on the flow network G (V, E) is defined as a real-valued function $f : V\times V \rightarrow R$ that satisfies the following properties:

1) *capacity constraint*: for all $(u,v)\in E$, $0\leq f(u,v)\leq c(u,v)$. This implies that the traffic rate at a link should not exceed the capacity of this link.
2) *skew symmetry*: if $(u,v)\in E$ and $(v,u)\notin E$, define $f(v,u)=-f(u,v)$; if $(u,v)\notin E$ and $(v,u)\notin E$, define $f(u,v)=0$.
3) *flow conservation constraint*: for all $u\in V-V_s\cup V_d$, $\sum_{v\in V} f(u,v)=0$. This implies that, for a node u that is neither in V_s and nor in V_d, the aggregate traffic rate entering u should be equal to the aggregate traffic rate leaving u.

To formulate the CA problem into a network flows problem, all the approaches in this category assume the traffic rates (either end-to-end or at each source) are known. In this subsection, we examine three such approaches: (1) Load Aware Channel Assignment [Raniwala *et al.* 2004], (2) Balanced Static Channel Assignment & Packing Dynamic Channel Assignment [Kodialam and Nandagopal 2005], and (3) Joint Routing and Channel Assignment and Link Scheduling [Alicherry *et al.* 2005].

To ease the examination of these three approaches individually, we first provide some of their similarities and differences. Briefly, they have the following two common characteristics.

- Their formulated problems are NP-hard, so all of them propose heuristic or constant-factor approximation methods to solve their problems.
- All of them combine their CA algorithms with routing algorithms. As explained in Section 2, channel assignment and routing are mutually dependent. Specifically for the network flows approaches, the routing algorithms decide the traffic rate at each link, which affects the CA decisions. On the other hand, the CA decisions determine the capacity of each link in the network, which affects the routing decisions. Thus, by combining these two algorithms, the network flows approaches aim to improve the network performance.

Though these three approaches all combine the CA and routing algorithms, they differ in that the first approach uses a loop consisting of the CA algorithm followed by the routing algorithm to make these two algorithms gradually converge to the final results, while the last two approaches jointly address the CA and routing problems by solving a Linear Program (LP).

(1) Load Aware Channel Assignment (LA-CA)

A pioneering work in the centralized CA approaches, [Raniwala *et al.* 2004] proposes two algorithms for solving the channel assignment problem: *Neighbor Partitioning Scheme (NPS)* and *Load-Aware Channel Assignment (LA-CA)*. The NPS is a very simple algorithm that partitions the neighbors at each node into n groups, where n is the number of radios at each node and can vary at each node. Its goal is that a node communicates with each group of its neighbors using only one radio and one channel, and the way of assigning channels reduces the interference in the network. On the other hand, the LA-CA is a complex algorithm that uses the network flows formulation. We first briefly describe the NPS algorithm because it is an early work and gives the intuition for solving the CA problem, and then detail the LA-CA algorithm as our focus.

In NPS, the CA problem is viewed as two subproblems: neighbor-to-interface binding and interface-to-channel binding. Neighbor-to-interface binding determines through which interface a node communicates with each of its neighbors, and interface-to-channel binding determines which channel is assigned to each of the interfaces. To complete these two bindings at each node, NPS randomly starts with one node, say v_1, and partitions v_1's neighbors into n groups, where each group has approximately equal number of neighbors. Then, each group is assigned to one of v_1's radios and each radio is assigned a currently least-used channel in terms of the number of radios using this channel within the interference range. In turn, each of v_1's neighbors performs the same procedure, while maintaining the following constraint: the radio used to communicate with v_1 must be assigned the same channel as the one used by v_1 to communicate with this neighbor. This process is iterated until all nodes have partitioned their neighbors. Fig. 8 gives an example channel assignment of a grid mesh network using this algorithm. In this figure, each node has two radios and four channels are used, numbered 1, 2, 3, and 4 respectively.

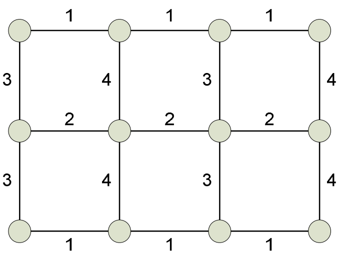


Fig. 8. An example channel assignment of a grid mesh network by neighbor partitioning

Next, the LA-CA algorithm is detailed in-depth. First, the inputs and outputs of its problem formulation are summarized as follows:

- The unit disk graph of the network is known.
- In the unit disk graph, the capacity of each link *l*, which can use multiple channels concurrently, is determined by (1) the number of non-overlapping channels assigned to *l* and (2) *l*'s bandwidth share at each of these non-overlapping channels, where *l*'s bandwidth share at a channel equals to the bandwidth of this channel divided by the number of links sharing this channel within *l*'s interference range.
- The end-to-end traffic rates (i.e., traffic loads) for a set of node pairs are known *a priori*.
- The output objective for CA is to ensure that each link's capacity is no less than the aggregate traffic load on it from every node pair.

After formally proving the above formulated CA problem is NP-hard, the LA-CA algorithm uses a heuristic approach to solve it. As mentioned previously, channel assignment and routing are mutually dependent. To break this mutual dependency, LA-CA starts with an estimation of the initial load on each link based on the *a priori* end-to-end traffic rates regardless of the channel assignment, and then iterate the channel assignment step followed by the routing step until the output objective stated above is achieved. The basic steps of the LA-CA algorithm are summarized as follows (see Fig. 9).

1. Initial Link Load Estimation: the initial traffic load on each link is estimated according to the given end-to-end traffic rates regardless of the channel assignment.
2. Channel Assignment: the links are visited in the decreasing order of link criticality, which is measured by the amount of traffic load on a link. When a link *l* is visited, it is greedily assigned a channel that has the least degree of interference, which is measured by the sum of the traffic loads on the links that share the same channel within *l*'s interference range.
3. Link Capacity Calculation: calculate the capacity of each link according to the CA results obtained in the last step.
4. Routing: with the new link capacities, the routing algorithm is executed to produce new traffic load on each link. According to its paper [Raniwala *et al.* 2004], the routing algorithm is not tied to the LA-CA and can be any routing algorithm designed for wireless networks. In [Raniwala *et al.* 2004], two routing algorithms are used in their simulations: one shortest path routing algorithm and one multi-path routing algorithm.
5. Comparison: if the capacity of each link is no less than the traffic load on it, the algorithm ends with the current CA and routing results. Otherwise, the current link load information is fed back to step 2 and the steps 2-5 are repeated.

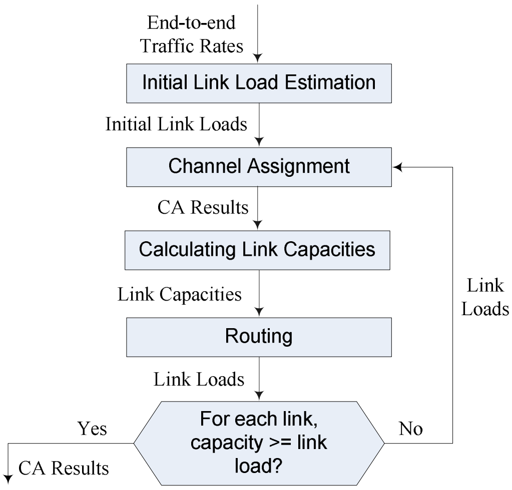


Fig. 9. Flowchart of the combined LA-CA and routing algorithms

The main advantages of the LA-CA algorithm are that (1) it uses a simple greedy heuristic such that it is efficient to implement and (2) its practicality is demonstrated in an IEEE 802.11b testbed. Its main limitation is that, as acknowledged in [Raniwala *et al.* 2004], it may fail to find feasible CA result due to the adoption of a simple greedy heuristic.

(2) Common backgrounds for the remaining two approaches

Since the remaining two approaches in this category both resort to an LP to solve the formulated network flows problems, their common backgrounds are presented here together to facilitate their respective descriptions.

First of all, both of them incorporate the various CA inputs and the objective of CA output into an LP. The CA inputs mainly include two parts: (1) the inputs related to the definition of a *flow* such as the end-to-end traffic rates and the *capacity constraint* and (2) the inputs arising from the M2WMN environment such as the node deployment, the number of radios at each node, the number of channels available to each radio, and the interference model. And the objective of CA output, such as maximizing the overall network throughput or minimizing the overall network interference, is expressed by the objective function of the LP.

Moreover, both of them propose algorithms for jointly solving the routing, channel assignment, and link scheduling problems in M2WMNs based on their formulated LPs. Two clarifications on this statement are given here. First, ‘jointly’ means that the three problems of routing, CA, and link scheduling are solved within a single formulation, not that they are solved simultaneously. Actually it can be seen from the later description that, for both of them, the routing problem is solved first to produce the routes for the input *flow*, and then the channel assignment and link scheduling problems are solved based on these routes. For the channel assignment and link scheduling problems, however, they may be solved either simultaneously or sequentially as to be described further. Second, in their LP contexts, these three problems have the following specific meanings:

- Routing: finding routes for the given *flow* in the given *flow network* G (V, E) such that the constraints included in the LP are satisfied.
- Channel Assignment: assigning channels to each link in E such that the obtained routes are fulfilled under the capacity constraint.
- Link Scheduling: finding the schedule of transmissions for links assigned to the same channel within the interference range, such that no interference occurs and the traffic rates of all routes can be accommodated. Note that, to make link scheduling possible, both approaches assume that all nodes in the network operate in a time-synchronized manner.

(3) Balanced Static Channel Assignment (BSCA) & Packing Dynamic Channel Assignment (PDCA)

Ref. [Kodialam and Nandagopal 2005] formulates the joint routing, CA, and scheduling problem in M2WMNs into a classic network flows problem — the *multi-commodity flow* problem [Ahuja *et al.* 1993]. In its formulation, a set of source nodes transmits to a set of destination nodes and the traffic between each source/destination pair is regarded as a commodity, with its end-to-end rate specified by an element in a rate vector. The LP framework proposed in [Kodialam and Nandagopal 2005] to solve this problem is flexible. It not only can specify a variety of network inputs such as channels with different bandwidth, but also can specify many linear objective functions such as determining the feasibility of a rate vector, maximizing the overall network throughput, or imposing certain fairness constraint.

Under this LP framework, the basic steps of the approach proposed in [Kodialam and Nandagopal 2005] to jointly obtain the routing, channel assignment, and link scheduling results are described as follows.

1. Determine the feasibility of the given rate vector and solve the routing problem.
2. Based on the routes obtained, both static and dynamic algorithms are proposed to solve the channel assignment and link scheduling problems. Here ‘static’ means the channel assignment is only decided once and remains unchanged in satisfying the rate vector, while ‘dynamic’ means the channel assignment can be changed at the beginning of certain time slot to respond to the traffic changes. Because calculating the optimal CA results is NP-hard, both algorithms use a greedy heuristic.
3. The static algorithm is named *Balanced Static Channel Assignment (BSCA)*, which introduces a concept called ‘constraint set’ and tries to minimize the maximum traffic rates on a constraint set when assigning a channel to a link. After the channel assignment, different time slots are assigned to the links that use the same channel within the interference range to avoid interference.
4. The dynamic algorithm is named *Packing Dynamic Channel Assignment (PDCA)*, which defines a time period consisting of n time slots (n>=1) and perform the channel assignment at the beginning of each time period. Note that n=1 means that the channels are reassigned at each time slot. PDCA uses a greedy packing-based heuristic, in which the links are assigned channels in the descending order of their traffic loads, with each assignment picking the channel that can currently provide the highest bandwidth share. Unlike BSCA, where channel assignment and link scheduling problems are solved sequentially, PDCA solves these two problems simultaneously.

As a summary to the BSCA&PDCA algorithms, we point out their advantages and limitations as follows.

- Advantage: the adopted LP framework is flexible so that it can incorporate a variety of inputs and objective output functions.
- Limitation: both BSCA and PDCA algorithms use a greedy heuristic, such that there is no performance bound in the worst case.

(4) Joint Routing, Channel Assignment, and Link Scheduling (RCL)

Overcoming the above limitation of the heuristic BSCA&PDCA algorithms, [Alicherry *et al.* 2005] proposes a constant-factor approximation algorithm to jointly solve the routing, channel assignment, and scheduling problems. However, the LP framework in [Alicherry *et al.* 2005] cannot incorporate as flexible inputs and outputs as that in BSCA&PDCA. Specifically, [Alicherry *et al.* 2005] considers the following inputs and outputs: (1) each node has its own number of radios and there are altogether K non-overlapping channels; (2) all the network traffic is to or from the Internet, which is represented by a single node $t \in V$ in its flow network G (V, E); (3) each mesh router $u \in V$ has an individual traffic rate to t, represented by $l\ (u)$; (4) the output objective is to maximize an identical coefficient λ for every node u such that the multiplied traffic rate $\lambda \cdot l(u)$ can be routed through the flow network. Note that λ is used to impose a fairness constraint, since the main concern of [Alicherry *et al.* 2005] is achieving fairness among all the nodes rather than maximizing the overall network throughput.

Ref. [Alicherry *et al.* 2005] names its approach to solve the above formulated problem *joint routing, channel assignment, and link scheduling (RCL)* algorithm collectively. With its common backgrounds with BSCA&PDCA described previously, we summarize the basic steps of RCL as below, with the CA step as our emphasis.

1. Solve LP: To overcome the NP-hardness, this step solves a LP relaxation of the formulated problem, obtaining the routes in the form of a flow $f : V \times V \rightarrow R$.
2. Channel Assignment: This step is based on the observation that the channel assignment is easy to perform if the number of available channels is equal to the minimum number of radios of every node (represented by I_{min}), since in this case each node can then be assigned all the channels. Accordingly, this step exploits a flow network transformation technique: it first transforms a flow network G into G' where each node has only I_{min} radios by creating multiple copies of each node; then, it assigns the channels on G' using only I_{min} channels out of the total K channels available; next, it revises the channel assignment by using the remaining K - I_{min} channel; finally, it maps the channel assignment back from G' to G.

3. Post Processing: Based on CA from the last step, this step readjusts the flow on the flow network, making efforts to minimize the maximum interference over all channels.
4. Flow Scaling: This step readjusts the flow and CA to make the interference-free link scheduling possible.
5. Interference-Free Link Scheduling: This step schedules the links sharing the same channel within the interference range to use different time slots, while still capable of carrying the flow.

As a summary to the RCL algorithm, we point out the main advantage of RCL is that it is the only constant-factor approximation algorithm proposed among the centralized approaches that strive to solve an NP-hard CA problem, while all other approaches adopt a heuristic algorithm providing no performance bound. And its main limitation is that the adopted LP framework is inflexible, which can only consider the simple inputs where the traffic rate from each node to the Internet is specified, and a single output objective of maximizing λ such that $\lambda\, l\,(u)$ for each node u has a feasible route.

(5) Summary to the network flows approaches

As a summary to the network flows approaches, we compare and contrast the three algorithms surveyed above based on their inputs, outputs, and heuristic/approximation methods in Table II.

Table II. Comparison and contrast of the three network flows approaches

	Inputs	Outputs	Heuristic/Approximation
LA-CA	• Node set and link set • Number of radios at each node and total number of channels • End-to-end traffic rates of a set of node pairs • Protocol Model as interference model	Channel assignment such that each link’s capacity is no less than its traffic load.	• The links are visited in the decreasing order of their amount of traffic loads • Greedily assign a channel that has the least degree of interference.
BS CA & PD CA	• Node set and link set • Number of radios at each node and total number of channels • A vector of end-to-end traffic rates • Protocol Model as interference model	Flexible CA objectives: many linear objective functions related to traffic load and link capacity can be achieved.	• BSCA greedily picks a channel that minimizes the maximum traffic rates on a constraint set • PDCA greedily picks a channel that can currently provide the highest bandwidth share.
RCL	• Node set and link set, with a special node t representing the Internet • Number of radios at each node and total number of channels • Source traffic rates from each node to the Internet • Protocol Model as interference model	Channel assignment that maximizes λ such that $\lambda\, l\,(u)$ for each node u can be routed.	• A constant-factor approximation method • Exploit a flow network transformation technique

The advantages and limitations of the network flows approaches as a category are summarized as follows.

- Advantage: the traffic load information, which is neglected by graph-based approaches, is inherently included in the network flows formulation.
- Limitation: all the three surveyed approaches assume constant traffic rates, which is not realistic in most practical networks where the traffic pattern is usually bursty and characterized by random on/off sources.

4.1.3 Network Partitioning Approaches. In network partitioning approaches, the CA problem is viewed as the problem of partitioning the radios and links in an M2WMN into disjoint subnetworks such that each subnetwork only uses a single channel and different channels are assigned to these subnetworks to reduce interference. Of all the surveyed centralized approaches, only [Brzezinski *et al.* 2006] falls in this category.

Specifically, [Brzezinski *et al.* 2006] assumes the following inputs from the M2WMNs:

- The *unit disk graph* G_N (V_N, E_N) of the network is given, called *network graph* in [Brzezinski *et al.* 2006].
- Each node $v \in V_N$ has a certain number of radios, and there are k channels available to the network.
- Interference constraint is given by the conflict graph G_I (V_I, E_I), also called *interference graph* in [Brzezinski *et al.* 2006].
- For each link $(i, j) \in E_N$, packets arrive in a stochastic process with an average rate λ_{ij}, which is specified in an arrival rate vector $\Lambda = (\lambda_{ij}, (i, j) \in E_N)$. Note that different from previously-surveyed approaches, [Brzezinski *et al.* 2006] uses stochastic processes instead of fixed traffic rates to describe the traffic load information.

As to the network partitioning output, [Brzezinski *et al.* 2006] does not partition an M2WMN arbitrarily. Instead, it requires the network topology of each obtained subnetwork to satisfy the following objective: the interference within this subnetwork due to the single channel can be avoided by an efficient distributed link-scheduling algorithm. To make this objective rigorous, [Brzezinski *et al.* 2006] defines the following two concepts (only their basic meanings are given here):

- *Stability region* (also called *capacity region*): the set of rate vector Λ that should be admitted by a properly-designed distributed link-scheduling algorithm.
- *Stable scheduling algorithm*: an algorithm that can admit any Λ in the stability region. Note that such an algorithm is also referred to as ‘an algorithm that can achieve 100% throughput’ in [Brzezinski *et al.* 2006].

For these two concepts, it is known that not all network topologies can have a stable scheduling algorithm. To find out what kinds of topologies can have such an algorithm, [Brzezinski *et al.* 2006] resorts to a recent work [Dimakis and Walrand 2006], which gives the sufficient conditions for the ‘maximal weight scheduling algorithm’ to be a stable algorithm and refers to these sufficient conditions as the ‘Local Pooling (LoP)’. Note that the maximal weight scheduling algorithm can be implemented as a distributed link-scheduling algorithm. To exploit the maximal weight scheduling algorithm as the stable scheduling algorithm used in each single-channel subnetwork, [Brzezinski *et al.* 2006] proves that a forest topology satisfies the LoP. Thus, [Brzezinski *et al.* 2006] formulates its network partitioning problem (i.e., the CA problem) as follows: given the inputs aforementioned, assign channels to each $(i, j) \in E_N$ (i.e., partition the M2WMN network into subnetworks) such that each subnetwork is assigned a single channel and has a forest topology.

Ref. [Brzezinski *et al.* 2006] solves this problem by reducing it to the *matroid intersection problem*, which has known algorithms to exploit. Before detailing the solution presented in [Brzezinski *et al.* 2006], some necessary background on *matroid* is given below.

A concept in combinatorics, *matroid* is defined as a pair (*E, I*), where *E* is a finite set and *I* is a set of subsets of *E*, with each element in *I* called an *independent set* of *E* that satisfies the following properties:

1) The empty set is independent, i.e., at least one subset of *E* is independent.
2) Every subset of an independent set is independent.
3) If A, $B \in I$ and *A* has more number of elements than *B*, then there exists an element $e \in A$ and $e \notin B$ such that $\{e\} \cup B$ is an independent set.

And the *matroid intersection problem* is stated as follows: given two matroid $M_1 = (E, I_1)$ and $M_2 = (E, I_2)$, find the largest independent set $A \in I_1 \cap I_2$ of *E*. For detailed information on matroid, the classic text [Oxley 1992] is recommended.

To perform the problem reduction, [Brzezinski *et al.* 2006] first introduces two matroids: graphic matroid and partition matroid, which incorporate the information contained in the inputs from an M2WMN. Then, [Brzezinski *et al.* 2006] reduces the formulated network partitioning problem into the following matroid intersection problem: finding the largest independent set in the intersection of the graphic matroid and the partition matroid. For matroid intersection problems, they can be solved optimally by a known polynomial time algorithm called ‘Matroid Cardinality Intersection (MCI)’ [Lawler 1976]. However, the forest topologies obtained by MCI vary considerably in size, so their stability regions also vary significantly, thus degrading the overall network throughput.

To balance the sizes of the forests and expand the stability region, [Brzezinski *et al.* 2006] proposes three ‘capacity expansion algorithms’ that perform further improvement using the results obtained by MCI as inputs and still retain the forest topology. These three algorithms are named ‘*greedy reallocation algorithm*’, ‘*maximum degree reallocation algorithm*’, and ‘*average degree reallocation algorithm*’ respectively, all of which are polynomial time algorithms. Since these three algorithms involve considerable mathematics, we only mention a basic fact about them here: the stability regions obtained by these three algorithms increase in the order as they are listed, and so do their complexities. Due to the above solution methodology, we refer to the algorithms proposed in [Brzezinski *et al.* 2006] collectively as Matroid Cardinality Intersection Channel Assignment (MCI-CA) for the convenience of later reference.

Two notes are worth making for MCI-CA. First, it needs to assume that the network operates in a time-synchronized manner such that the links in a subnetwork can be scheduled with the maximum weight scheduling algorithm. Second, according to the above description of MCI-CA, though [Brzezinski *et al.* 2006] is titled 'Enabling distributed throughput … ', MCI-CA is actually a centralized CA approach.

Finally, we point out the advantages and limitations of MCI-CA as follows.

- Advantages: (1) among all the surveyed centralized approaches, only MCI-CA formulates the CA problem into a polynomial-time optimization problem, not an NP-hard problem as in other approaches; (2) each forest obtained by partitioning can use a distributed link-scheduling algorithm to achieve the stability region.
- Limitations: (1) in case the number of channels is not large enough to assign each partition a unique channel, the stability region in some partitions can only be partially achieved; (2) the partitioning of network topology into forests may not be optimal in terms of overall network throughput.

4.2 Distributed CA Approaches

Since the distributed CA approaches involve communication and coordination among multiple parties, they are more difficult to design than their centralized counterparts. Accordingly, we only find four recent distributed CA approaches for survey in this paper.

In all these four distributed CA approaches, each node measures local channel statistics and exchanges them with other nodes to calculate the channel assignment. However, these approaches differ in their choices of the local channel statistics. For instance, these choices can be the number of links sharing a common channel within the interference range, the traffic load on a channel, the signal to interference and noise ratio on a channel, or a combination of them. For convenience, we refer to these channel statistics as CA metrics hereafter. Analogous to the routing metrics that serve as the basis for routing, the CA metrics also play an equivalent role in making the CA decisions. Therefore, when each CA approach is examined below, special attention is paid to its adopted CA metrics.

According to Section 3, the distributed CA approaches are classified into two categories: gateway-centered approaches and peer-oriented approaches, which are in turn explored in the following subsections.

4.2.1 Gateway-centered Approaches. Gateway-centered approaches assume a gateway as the central source and sink of the major network traffic. In this subsection, two such approaches are examined: Hyacinth [Raniwala and Chiueh 2005] and DMesh [Das *et al.* 2006].

(1) Hyacinth

As a pioneering work in distributed CA research, [Raniwala and Chiueh 2005] proposes an M2WMN architecture called 'Hyacinth', where the existence of a gateway is assumed and most network traffic is to or from the Internet via this gateway. In Hyacinth, both distributed CA and routing protocols are proposed. Since the CA protocol is based on the tree topology established by the routing protocol, the routing protocol is described first below. Basically, it finds paths for all network nodes to the gateway by establishing a tree topology rooted at the gateway. Following the main idea of the spanning tree protocol in IEEE 802.1D, this routing protocol operates as follows. To establish a tree, each node (say *A*) periodically broadcast an ADVERTISE packet to its neighbors, which mainly contains its *cost* to the gateway. Note that three options are defined for the *cost* metric in Hyacinth and its details are omitted here. When one of *A*'s neighbors receives this ADVERTISE packet, it compares its current cost to the gateway with *A*'s advertised cost plus its cost to *A*. If it sees a 'larger than' relationship, it uses *A* as its parent to reach the gateway, otherwise not. As a consequence of this routing protocol, each node will know its parent and children in the resultant tree topology. To enable fault-tolerance, each node is also made to store a backup parent.

Next, the CA protocol in Hyacinth is described in-depth as follows. Originating from the centralized NPS algorithm (see Section 4.1.2) proposed by the same authors, this distributed CA protocol also views the CA problem as two subproblems: neighbor-to-interface binding and interface-to-channel binding. In neighbor-to-interface binding, each node's network interface cards (NICs) are divided into two classes: (1) UP-NICs that are used to connect to the parent and (2) DOWN-NICs that are used to connect to the children. With this division, the binding of neighbors and interfaces is determined. In interface-to-channel binding, each node only needs to assign channels to its DOWN-NICs, and mandates each of its UP-NICs to use the same channel as the one used by its parent's corresponding DOWN-NIC (see Fig. 10). In order to select a channel for a DOWN-NIC, each node measures the traffic loads on all its channels periodically and exchanges this information via the CHNL-USAGE packets with other nodes in its interference range. Then, each node can calculate the metric for making the CA decisions — the *total load* of a channel, which is a weighted sum of two measures: (1) the total number of links using this channel within the interference range and (2) the aggregate traffic load on this channel from all links using this channel within the interference range. After calculating the *total load*s of all its channels, a node orders these channels according to their *total load*s, thus obtaining the least-used channel.

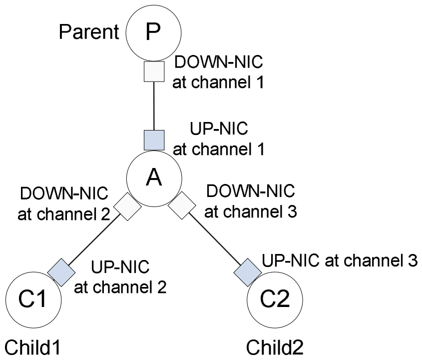


Fig. 10. An example of CA in Hyacinth

Besides determining the least-used channel, every node is also given a priority to differentiate their privileges to use channels. Specifically, the priority of a node is defined as its hop distance to the gateway, with a smaller number of hops indicating a higher priority. By combining the total load metric and the priority mechanism, the CA operation in Hyacinth can be summarized as follows: in assigning a channel to a DOWN-NIC, a node always chooses the least-used channel that is not currently used by a higher priority node within the interference range. The goals of this CA operation are twofold: reducing the local interference at a node and offering the near-gateway links higher priorities for choosing channels and hence higher bandwidth.

As a note, the rationale for the two-step bindings in Hyacinth is to prevent the ripple effect. By mandating that each node is only responsible for the channel assignment at the DOWN-NICs, the propagation of consecutive channel switching is blocked. We view this neat solution of ripple effect as a major advantage of Hyacinth. And the major limitation of Hyacinth is that when a new node joins the network, a long period of channel scanning process is needed to discover the DOWN-NIC channels of its potential parents. This process has a length from 5 to 10 seconds as mentioned in Hyacinth, during which a new node broadcasts HELLO packets at all channels to perform the discovery.

(2) DMesh

While Hyacinth pioneers the gateway-centered distributed CA approaches, [Das *et al.* 2006] constitutes a further advance within this category that considers directional antennas. Specifically, [Das *et al.* 2006] proposes an M2WMN architecture called 'DMesh', where each node has two types of radios: (1) one radio with an omni-directional antenna to transmit control packets as well as data packets in case of fault recovery and (2) multiple additional radios with directional antennas to transmit data packets. The goal of DMesh is to exploit both spatial separation by directional antennas and frequency separation by non-overlapping channels to improve the network capacity.

Similar to Hyacinth, DMesh proposes both distributed routing and CA protocols. Its routing protocol modifies the Optimized Link State Routing protocol (OLSR) [Clausen and Jacquet 2003] in order to exploit the directional antennas. Accordingly, it is named 'Directional OLSR (DOLSR)'. Inspired by the routing protocol in Hyacinth, DOLSR also establishes a tree topology over the network with a gateway as the root and maintains the parent-children relationship between the nodes along this tree.

To be cost-effective, DMesh does not adopt the directional antennas with perfect spatial separation due to their high price. Instead, the directional antennas in DMesh have the interference range in the shape of a cone, and a 45° degree of beamwidth is assumed in the simulations. Thus, a CA protocol is still needed to reduce the interference within the interference cones. In general, CA in DMesh requires every node maintaining a 'channel map' composed of the following three state vectors:

- Channel Vector: records whether it uses each channel.
- Rate Vector: records its traffic rates at each channel.

- Distance Vector: records the directional information of the neighbor nodes that it communicates with at each channel.

Every node exchanges this channel map with its interfering neighbors in consideration of directionality, thus exchanging both the traffic and the directional information. Similar to Hyacinth, the CA protocol in DMesh also mandates that each node is only responsible for assigning channels to its children. In performing such an assignment, four directional CA schemes are proposed. Limited by the purpose of this survey, only their basic idea is given as follows. When a node X attempts to assign a channel to one of its children Y, it first looks for a free channel that is not used by any node whose interference cone contains X or Y. If such a channel exists, this channel is assigned to Y. Otherwise, X looks for a channel that is the least-loaded in terms of the aggregate traffic rate on this channel (or simply, aggregate number of traffic flows) and assigns it to Y. Note that, unlike Hyacinth, there is no priority mechanism in DMesh to favor near-gateway links in assigning channels. This is perhaps because the spatial separation by directional antennas can provide the near-gateway links enough capacity.

As a summary, the advantages and limitation of DMesh are pointed out as follows.

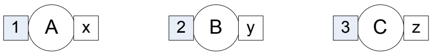
- Advantages: (1) both spatial separation and frequency separation are exploited to reduce interference, such that a significant throughput gain, contrasted with an omni-directional solution (termed ‘OMesh’ in DMesh), is achieved as shown in their simulations; (2) the practicality of DMesh is validated with a 16-node testbed that uses commercial directional antennas.
- Limitation: as mentioned in DMesh, the directions of the antennas are manually positioned during network deployment and remain unchanged during network operation. Thus, a CA protocol that considers steerable directional antennas is more appealing.

4.2.2 Peer-oriented Approaches. Unlike the gateway-centered approaches, the peer-oriented approaches do not make any assumptions on the traffic pattern, so they are general to accommodate different kinds of peer-to-peer traffic. In this subsection, two such approaches are examined: (1) Probabilistic Channel Usage based Channel Assignment [Kyasanur and Vaidya 2006] and (2) Joint Optimal Channel Assignment and Congestion Control [Mohsenian Rad and Wong 2006].

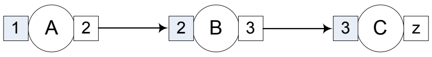
(1) Probabilistic Channel Usage based Channel Assignment (PCU-CA)

To prevent the ripple effect, [Kyasanur and Vaidya 2006] also classifies the interfaces at a node into two classes: *fixed interfaces* and *switchable interfaces*. The channels used by the former are changed periodically by the CA protocol but stay fixed during each period, while the channels used by the latter can change at any time according to the need of network traffic. Note that the channels used by the fixed interfaces are simply called ‘fixed channels’ in [Kyasanur and Vaidya 2006].

The basic idea of this interface classification is that the fixed interfaces are used to receive network traffic by staying at the fixed channels, while the switchable interfaces are used to transmit network traffic by switching to the channels of the fixed interfaces. Of course, if the fixed interfaces at two nodes happen to share the same channel, they communicate with this fixed channel. Because the channels of every node’s fixed interfaces stay unchanged in each period, the ripple effect is blocked. An example of protocol operation with the above idea is illustrated in Fig. 11, where each of the three depicted nodes A, B, and C exactly has one fixed interface (indicated by a gray shade) and one switchable interface. Initially, the fixed interfaces of A, B, and C use channel 1, 2, and 3 respectively. At certain time, assume A has traffic to send to C via B. To accomplish this transmission, A’s switchable interface (originally at channel x) switches to channel 2 so as to transmit to B, and then B’s switchable interface (originally at channel y) switches to channel 3 so as to transmit to C, resulting in Fig. 11 (b).



(a) Initially



(b) After Switching

Fig. 11. An example of protocol operation with fixed interfaces and switchable interfaces

Corresponding to the two classes of interfaces, the CA protocol proposed in [Kyasanur and Vaidya 2006] has two parts: assigning channels to the fixed interfaces and switching channels at the switchable interfaces, which are described below respectively. The CA metric used in the first part is called the 'channel usage', which is measured by the number of nodes using this channel within a two-hop neighborhood (i.e., the assumed interference range). In the protocol operation of the first part, each node maintains two data structures: a NeighborTable that records the fixed channels of its one-hop neighbors, and a ChannelUsageList that records the channel usage of each channel. With these two data structures, the channel assignment for the fixed interfaces operates in the following steps.

1. At the network initialization, each node randomly selects channels for its fixed interfaces.
2. Each node periodically broadcasts Hello packets on all channels to its one-hop neighbors. A Hello packet mainly contains the fixed channels and the current NeighborTable of the sender.
3. Based on the Hello packets received from its neighbors, each node updates its NeighborTable and ChannelUsageList. Since the NeighborTables from its neighbors contain one-hop channel usage information, the ChannelUsageList can assemble two-hop channel usage information.
4. According to its ChannelUsageList, a node decides whether the number of other nodes using one of its fixed channels is large (a configurable criterion). If so, it changes this channel to a less-used channel with a probability 0.4 (also configurable). This probabilistic channel switching is to prevent the *channel oscillation*, an undesirable phenomenon described in Section 2.

The second part of the CA protocol requires each node to maintain packet queues for each channel with packets to transmit. When a node has packets to transmit to a neighbor, it looks up the NeighborTable to obtain the fixed channel of this neighbor, and then inserts the packets to the corresponding channel queue. The fixed interface is responsible for transmitting packets in the queue of the fixed channel, while the packets in other queues are transmitted by the switchable interfaces. A switchable interface changes its channel to transmit packets when either of the following two conditions is true: (1) it is on a channel with an empty queue and (2) it has been on a channel for a MaxSwitchTime duration. The rule for selecting a new channel is that the new channel queue contains the currently earliest packet waiting to transmit, thus providing fairness to different channel queues.

As seen above, the channels used in transmission actually depend on the channels assigned to the fixed interfaces, so the first part of the CA protocol determines the channel assignment. Since the first part CA protocol assigns channels based on the channel usage metric in a probabilistic manner, we refer to the entire CA protocol, which is not named in [Kyasanur and Vaidya 2006], as Probabilistic Channel Usage based Channel Assignment (PCU-CA) for the convenience of later reference.

As stated in [Kyasanur and Vaidya 2006], PCU-CA is implemented in the link layer, and its implementation is independent of the upper layers such that any routing protocol can run on top of it. To have a better combination with PCU-CA, [Kyasanur and Vaidya 2006] also proposes a routing protocol that is capable of considering the channel switching delays in the link layer, which happen frequently in PCU-CA (especially at the switchable interfaces). Limited by the scope of this paper, the details of the routing protocol are omitted here.

As a summary to PCU-CA, its advantages and limitations are given as follows.

- Advantages: (1) both ripple effect and channel oscillation are tackled, thus stabilizing the network; (2) the 'channel usage', the number of nodes sharing this channel within two hops, is easy to obtain, thus simplifying the protocol.
- Limitations: (1) despite prevention of channel oscillation, the probabilistic channel switching may not be optimal for the network performance; (2) the traffic load information is not considered, so the criticality of links is ignored in assigning channels.

(2) Joint Optimal Channel Assignment and Congestion Control (JOCAC)

Different from all other CA approaches surveyed in this paper, [Mohsenian Rad and Wong 2006] views the CA problem from a novel perspective: optimizing the performance of TCP congestion control. The rationale for this idea is as follows: if CA does not perform well, interference will cause severe congestion in wireless links, which degrades the TCP performance significantly due to the AIMD algorithm adopted by TCP congestion control. Accordingly, [Mohsenian Rad and Wong 2006] formulates the CA problem into the Joint Optimal Channel Assignment and Congestion Control (JOCAC) problem and names its proposed algorithm to solve this problem the JOCAC algorithm. Basically, the JOCAC problem is a distributed utility maximization problem for congestion control with the interference modeled by the Physical Model. Below, some background on the JOCAC problem is given first, and then the JOCAC algorithm is described in depth.

The distributed utility maximization for congestion control is a general analytic model proposed in [Kelly 1997], which views the network as a set of L links, with each link having capacity c_l ($l \in 1, 2, \ldots, L$). There are S sources in the network, and each source s has a transmission rate r_s ($s \in 1, 2, \ldots, S$) and a utility function $U_s(r_s)$. To model the congestion control feedback, each link is associated with a congestion price λ_l, and each source is assumed to have the access to the aggregate price $q_s(r_s)$ of all the links in its route to the destination. Each source *s* tries to maximize its profit: $U_s(r_s) - q_s(r_s)$, while the role of congestion price is to adjust the behavior of each source such that the aggregate utility of all sources ($\sum_{s\in S} U_s(r_s)$) is maximized, subject to the link capacity constraint that the aggregate traffic rate from all sources at each link does not exceed the capacity of this link. A key restriction of the above model is that each source or link only needs to measure its local information, such that the model can be implemented distributedly. Up to now, many analytical works on TCP congestion control have been done using this model. $U_s(r_s)$ and λ_l for well-known TCP congestion control versions Reno and Vegas are well defined, as summarized in the excellent review paper [Low *et al.* 2002].

Based on the above analytic model and existing definitions of $U_s(r_s)$ and λ_l for TCP Vegas, [Mohsenian Rad and Wong 2006] extends these results by incorporating into them the M2WMN parameters such as the number of available channels, number of radios at each node, and SINR at each channel. Thus, the new mathematical expressions of $U_s(r_s)$ and λ_l in the M2WMN environment are obtained, and the JOCAC algorithm that maximizes $\sum_{s\in S} U_s(r_s)$ subject to the link capacity constraint is derived within the new formulation. Involving considerable mathematical calculations, the details of the JOCAC algorithm are omitted here. Instead, the basic behavior of JOCAC's distributed implementation is summarized as follows: each node periodically measures and exchanges with other nodes the local information that mainly includes the SINR and λ_l, and then calculates the CA result using the JOCAC algorithm based on the information measured locally and received from other nodes.

Three notes on the JOCAC algorithm are made below. First, no routing protocol is proposed in [Mohsenian Rad and Wong 2006], which assumes that the routes are determined by existing routing protocols. Second, as mentioned in [Mohsenian Rad and Wong 2006], its JOCAC algorithm can also be implemented in a centralized manner, in which a central node is needed to gather the information, execute the algorithm, and distribute the CA result. Third, though the JOCAC algorithm is compared with the CA approach in Hyacinth (a gateway-centered approach) with a tree topology network in its simulation, JOCAC is a peer-oriented approach according to the basic behavior of its distributed implementation. This is further verified by its simulation setup, where the 30 TCP Vegas flows randomly select its destinations among the mesh nodes, instead of using the Internet as its destination.

As mentioned in [Mohsenian Rad and Wong 2006], its distinguishing advantage is that it can utilize not only the non-overlapping channels but also the partially overlapping channels[3]. This is due to its modeling of interference using SINR, which means that if the SINR is larger than the capture threshold, the packets can be received correctly. That is, the partially overlapping channels can be used, as long as the interference caused by the overlapping frequency bands is not significantly strong. And its limitation is that it introduces a large amount of control traffic in its distributed implementation: each node periodically exchanges information such as SINR and λ_l with all other nodes, since the calculation of CA at each node requires this information along the route for each TCP flow. In this sense, the distributed implementation of the JOCAC algorithm is similar to the link state routing protocols, where the link states of each node are flooded to the entire network.

4.2.3 Summary to Distributed CA Approaches. As seen in the above discussion of the four distributed CA approaches, the CA metrics play an important role in assigning channels, so we compare the CA metrics adopted by them based on some basic characteristics in Table III. Note that the detailed definitions of these CA metrics are already given in the description of their corresponding approaches.

---

[3] For instance, in FCC regulation for IEEE 802.11b, there are 3 non-overlapping channels out of the 11 partially overlapping channels.

Table III. Comparison on the CA metrics adopted by the four distributed approaches

	Subcategory	Name of Metrics	Basic Channel Statistics Included	Complexity
Hyacinth	Gateway-centered	total load	(1) the number of links using this channel (2) traffic load	Medium
DMesh	Gateway-centered	channel map	(1) the number of links using this channel (2) traffic load (3) direction information	Medium
PCU-CA	peer-oriented	channel usage	(1) the number of links using this channel	Lowest
JOCAC	peer-oriented	SINR and congestion price	(1) signal to interference and noise ratio	Highest

Moreover, we highlight the advantages and limitations of the two subcategories of distributed CA approaches, gateway-centered and peer-oriented, as follows.

- Gateway-centered approaches: their advantage is the opportunity to utilize the gateway-centered traffic pattern to ease and optimize the CA task, and their limitation is the incapability of accommodating other kinds of traffic patterns.
- Peer-oriented approaches: their advantage is the capability of adapting to various kinds of traffic patterns, and their limitation is the difficulty of dealing with CA issues such as fault tolerance, ripple effect, and channel oscillation without any assumption on the traffic pattern.

4.3 Summary to All CA Approaches

In the previous two subsections, we have surveyed eleven state-of-the-art centralized and distributed approaches, with their advantages and limitations identified individually. As an overall summary to these approaches, we next present a comprehensive table that compares and contrasts them based on ten basic properties related to channel assignment (see Table IV). These properties are listed as follows, with their four-character labels indicated within parentheses:

1. A default common channel is used to transmit control messages. (DeCh)
2. Traffic load information is considered. (TrLd)
3. The existence of gateway nodes is required to facilitate CA. (GwNd)
4. The Physical Model is adopted as the interference model. (PhMo)
5. Ripple effect is addressed. (RpEf)
6. Channel oscillation is addressed. (ChOs)
7. Routing scheme is proposed in combination with CA. (Rtng)
8. Fault tolerance is supported. (FaTo)
9. Fairness is supported. (Fair)
10. Testbed implementation is conducted. (TBed)

The importance of these properties is noted as follows. For property 1, a default common channel provides convenience to address the connectivity, stability, and fault tolerance issues, but it incurs hardware overhead and consumes a frequency band. For property 2, considering traffic load information enables the CA results to adapt to the current network situation, but makes the CA operation complex. For property 3, exploiting gateway nodes eases the CA design at the price of sacrificing generality. Properties 4 – 9 are selected from the key design issues identified in Section 2, with their meanings described and their importance justified there. Property 10 is concerned with testbed implementation, which reflects the practicality of the corresponding CA approach. Note that two key design issues identified in Section 2 are not included in the above properties (i.e., connectivity and throughput/latency), since they are imperative for any CA approach to address, thus omitted.

Table IV. Comparison and contrast of all CA approaches

	1. DeCh	2. TrLd	3. GwNd	4. PhMo	5. RpEf	6. ChOs	7. Rtng	8. FaTo	9. Fair	10. TBed
CLICA	No	No	No	No	N/A	N/A	No	No	No	No
INSTC	No	No	No	No	N/A	N/A	Yes	No	No	No
BFS-CA	Yes	No	Yes	No	N/A	N/A	No	No	No	Yes
LA-CA	No	No	Yes	No	N/A	N/A	Yes	No	No	Yes
BSCA & PDCA	No	Yes	No	No	N/A	N/A	Yes	No	Yes	No
RCL	No	Yes	Yes	No	N/A	N/A	Yes	No	Yes	No
MCI-CA	No	Yes	No	No	N/A	N/A	No	No	Yes	No
Hyacinth	No	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes
DMesh	Yes	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Yes
PCU-CA	No	No	No	No	Yes	Yes	Yes	Yes	Yes	No
JOCAC	No	Yes	No	Yes	No	No	No	Yes	Yes	No

In Table IV, the names of the distributed approaches are shown in italic, so as to be distinguished from the centralized approaches. If an approach satisfies/dissatisfies a property, the corresponding table entry is marked by Yes/No respectively. If a property cannot be applied on an approach, the corresponding table entry is marked by N/A (Not Applicable). For example, due to the static nature of the centralized approaches, there are no ripple effect and channel oscillation issues in them, so all the corresponding entries are marked by N/A.

This table provides the following additional understandings to the current CA approaches besides the summarizations given in the previous subsections.

- Properties 4 and 6 are only addressed by a single approach each, which shows that the two corresponding issues — Physical Model and channel oscillation — receive very limited attention from the current CA research. Accordingly, these two issues are further discussed in the next section about future research directions.
- Property 8 is supported by all of the distributed approaches and none of the centralized approaches, which shows that the distributed approaches can adapt to the dynamic network changes while the centralized ones fail to possess such capability.
- According to the criterion on fairness for CA described in Section 2, if an approach explicitly shows the support of fairness or considers the traffic load while performing CA, we regard it as supporting fairness, otherwise not. As a result, eight of eleven approaches provide such fairness support, showing that fairness has received adequate attention from the current approaches.
- Testbed implementation is conducted by four out of eleven approaches, showing that the current research efforts are increasingly attaching importance to the practicality of their proposals. This trend is especially beneficial for the protocols in wireless environment, where the modeling of physical channels is still immature and only real-world implementations allow the protocols to be tested upon various channel uncertainties.

For the two broad categories of CA approaches (centralized and distributed), we also summarize their advantages, limitations, and usage scenarios in Table V.

Table V. Advantages and limitations of the centralized and distributed approaches

	Advantages	Limitations	Usage Scenario
Centra-lized	Capable of getting the optimal or near-optimal CA results, due to the availability of entire network information.	Assume stable nodes/links and static traffic pattern, due to the difficulty of gathering and distributing information globally in an M2WMN.	During network deployment or maintenance stages.

Distri-buted	Capable of quickly adapting to network changes and failures, because of only relying on the local information.	CA results may be far from global optimality, due to the use of only partial information.	During network operation stage.

5. FUTURE RESEARCH DIRECTIONS

As a new research area, CA for M2WMNs still needs to be complemented and improved to reach maturity. Also, with the fast development of wireless communication technologies, CA techniques will continue its evolution to keep up with the new changes. In this section, we first foresee the general research directions for CA, and then pinpoint some specific future research issues.

For the general research directions, we first argue that more research efforts should be put into the distributed CA approaches, because (1) the distributed approaches are needed by the practical M2WMN networks, which exhibit considerable network dynamics such as nodes/links failure, traffic load changes, and external interference and (2) all of the specific future research issues to be discussed below require further attention from the distributed approaches. A second general research direction is to consider more realistic physical layer in the CA design. As shown in Table IV, most current CA approaches only consider the Protocol Model as the interference model, which is not very realistic. We believe that using the signal strength instead of the predetermined interference range to decide the channel status is a more desirable method in future. Two signal strength metrics obtainable from the physical layer are the RSSI (Receive Signal Strength Indicator) suggested by the measurement study [Paul *et al.* 2007] and the SINR used in the JOCAC [Mohsenian Rad and Wong 2006].

Besides the above general directions, we also highlight the following specific issues, which are not fully addressed or remain unaddressed by the current CA approaches, to be tackled in future.

- External Interference: since the communication in M2WMNs uses the IEEE 802.11 and 802.16 standards that operate in the 2.4GHz and 5GHz unlicensed bands (802.16 can also use RF bands 10-66GHz), there is no guarantee that other external wireless sources do not use the same RF bands. In some of the current M2WMN deployments [BelAirNetworks 2007, Nortel 2006], the status of all employed channels is constantly monitored by the radios at each node, and if external interference is detected by a radio, the channel assignment mechanism at the node can dynamically switch this radio to a new channel. In the surveyed approaches, only [Ramachandran *et al.* 2006] considers the external interference. Unfortunately, it can only detect the interference from external wireless networks using the IEEE 802.11 standard, since its external interference monitoring relies on the identification of external MAC addresses. Therefore, more research efforts should be devoted to avoiding external interference in designing the CA approaches in future. We believe that the basic means to address this issue is to exploit the rich functionalities provided by the physical layer to retrieve the channel status.
- Directional Antennas: in the current M2WMN deployments [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005], directional antennas are used among mesh routers. For them, the directionality of the antennas is taken to be perfect, so the interference issue is simplified, only considering the assignment of different channels to the interfaces at a single node and disregarding the interference from the neighbor nodes. In the surveyed approaches, only DMesh [Das *et al.* 2006] considers directional antennas, but imperfect directionality is assumed, so it still addresses the interference within the interference cones. With the hardware price continuously decreasing, directional antennas with strict directionality will dominate in the future. Therefore, designing CA with the interference issue simplified by the directional antennas, which thereby has the opportunity to better addressing other issues, should be considered by future research.
- Channel Oscillation: as shown in Table IV, only PCU-CA [Kyasanur and Vaidya 2006] addresses this phenomenon. Recall that PCU-CA uses a probabilistic channel switching mechanism, which prevents the channel oscillation but may not be optimal for the network performance. Besides the probabilistic mechanism, the techniques used to solve the routing oscillation can also be borrowed with certain adaptation to tackle the channel oscillation. To name some such techniques, a hysteresis factor is used to avoid overlay route flap in [Zhao *et al.* 2003], and limiting route advertisements by fixed timers is used to damp BGP route flap in [Villamizar *et al.* 1998].
- Quality of Service (QoS): QoS is needed by some major WMN applications such as VoIP and video surveillance [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005]. Though no CA approaches discussed in this paper address QoS, some of them address QoS in their combined routing algorithms. Similarly, the current M2WMN deployments [BelAirNetworks 2007, Nortel 2006, TroposNetworks 2005] also provide QoS support in layer-2 frame forwarding or in layer-3

packet routing, where the frames/packets can be classified and thereby treated differently. This leads to the natural question whether CA is the proper place to implement QoS functionality. To this question, our answer is that CA should provide QoS support since it decides the capacity of network links, which is the fundamental resource for providing QoS. Though CA does not forward network packets directly, it should be designed to have QoS in mind, thus providing bandwidth allocation guarantee needed by the packet forwarding components in the layer-2/3 protocols. For example, one kind of QoS support CA can offer is to guarantee that the bandwidth of each link is no less than a certain minimum value.

6. CONCLUSION

With wireless communication probably the fastest developing technology today, the WMNs have swiftly evolved from a single-radio single-channel architecture to a multi-radio multi-channel architecture. Having been adopted in many WMN solutions commercially, this multi-radio multi-channel architecture gives rise to a new research area — channel assignment, which seeks to find the optimal mapping between the channels and radios at each node to improve the network performance.

In this paper, considerable effort is made to provide insight into the state-of-the-art approaches proposed in the CA research area. Our contributions are fourfold. Firstly, the key design issues for CA approaches are identified, with the rationale for their importance clarified. Secondly, a classification of the CA approaches is proposed, where the CA approaches are classified into two categories overall: centralized and distributed. The centralized approaches are further classified according to their problem formulations and the distributed approaches are further classified according to their assumed traffic patterns. Thirdly, each approach is treated with a description of its underlying idea and basic steps, followed by a remark pointing out its pros and cons. As a summary to all the approaches, a comprehensive comparison and contrast on them is made based on the basic properties mainly arising from the key design issues. Finally, both general and specific future research directions for channel assignment are highlighted, with solution hints given.

REFERENCES

AHUJA, R.K., MAGNANTI, T.L. and ORLIN, J.B. 1993. *Network flows : theory, algorithms, and applications.* Prentice Hall, Englewood Cliffs, N.J.

AKYILDIZ, I.F., WANG, X. and WANG, W. 2005. Wireless mesh networks: a survey *Computer Networks* 47 (4), 445-487

ALICHERRY, M., BHATIA, R. and LI, L.E. 2005. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. *ACM MobiCom*, (Cologne, Germany), 58-72.

BELAIRNETWORKS. 2007. Building Inclusive Municipal Wireless Mesh Networks. http://www.belairnetworks.com, White Paper.

BRUNO, R., CONTI, M. and GREGORI, E. 2005. Mesh networks: commodity multihop ad hoc networks. *IEEE Communications Magazine* 43 (3), 123- 131.

BRZEZINSKI, A., ZUSSMAN, G. and MODIANO, E. 2006. Enabling distributed throughput maximization in wireless mesh networks: a partitioning approach. *ACM MobiCom*, 26 - 37.

CLARK, B., COLBOURN, C. and JOHNSON, D. 1990. Unit disk graphs. *Discrete Mathematics* 86 (1), 165–177.

CLAUSEN, T. and JACQUET, P. 2003. Optimized Link State Routing Protocol (OLSR). *RFC 3626, Experimental.*

CORMEN, T.H. 2001. *Introduction to algorithms.* MIT Press, Cambridge, Boston, MA.

DAS, S.M., PUCHA, H., KOUTSONIKOLAS, D., HU, Y.C. and PEROULIS, D. 2006. DMesh: Incorporating Practical Directional Antennas in Multichannel Wireless Mesh Networks. *IEEE Journal on Selected Areas in Communications* 24 (11), 2028 - 2039.

DIMAKIS, A. and WALRAND, J. 2006. Sufficient conditions for stability of longest queue first scheduling: second order properties using fluid limits. *Advances of Applied Probabilities* 2 (38), 505-521.

GHOSH, A., WOLTER, D.R., ANDREWS, J.G. and CHEN, R. 2005. Broadband wireless access with WiMax/802.16: current performance benchmarks and future potential. *Communications Magazine, IEEE* 43 (2), 129-136.

GUPTA, P. and KUMAR, P.R. 2000. The capacity of wireless networks. *Information Theory, IEEE Transactions on* 46 (2), 388-404.

IEEE. 1999. Supplement to IEEE standard - Part 11: wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: high-speed physical layer in the 5 GHz band. *IEEE Std 802.11a-1999*.

IEEE. 2000. Supplement To IEEE Standard - Part 11: Wireless LAN Medium Access Control (MAC) And Physical Layer (PHY) Specifications: Higher-speed Physical Layer Extension In The 2.4 GHz Band. *IEEE Std 802.11b-1999*, 1-90.

IEEE. 2003. IEEE Standard for Local and metropolitan area networks - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *ANSI/IEEE Std 802.11, 1999 Edition (R2003)*, 1-513.

IEEE. 2004. IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems. *IEEE Std 802.16-2004 (Revision of IEEE Std 802.16-2001)*, 1-857.

JAIN, K., PADHYE, J., PADMANABHAN, V.N. and QIU, L. 2003. Impact of Interference on Multi-Hop Wireless Network Performance. *ACM MOBICOM*, (San Diego, CA, USA), 66-80.

KELLY, F.P. 1997. Charging and rate control for elastic traffic. *European Transactions on Telecommunication* 8, 33-37.

KODIALAM, M. and NANDAGOPAL, T. 2005. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. *ACM MobiCom*, 73 - 87

KYASANUR, P. and VAIDYA, N.H. 2006. Routing and Link-layer Protocols for Multi-Channel Multi-Interface Ad Hoc Wireless Networks. *ACM SIGMOBILE Mobile Computing and Communications Review* 10 (1).

LAWLER, E.L. 1976. *Combinatorial optimization : networks and matroids.* Holt, Rinehart and Winston, New York.

LOW, S.H., PAGANINI, F. and DOYLE, J.C. 2002. Internet congestion control. *Control Systems Magazine, IEEE* 22 (1), 28-43.

MARINA, M.K. and DAS, S.R. 2005. A Topology Control Approach for Utilizing Multiple Channels in Multi-Radio Wireless Mesh Networks. *The 2nd International Conference on Broadband Networks*, 381 - 390.

MESHDYNAMICS. 2006. Technology - Performance Analysis. http://www.meshdynamics.com, Online.

MOHSENIAN RAD, A.H. and WONG, V.W.S. 2006. Joint Optimal Channel Assignment and Congestion Control for Multi-channel Wireless Mesh Networks. *IEEE ICC'06*, 1984-1989.

NORTEL. 2006. Nortel Municipal Wireless Solutions. http://www.nortel.com, Solution Brief.

OXLEY, J.G. 1992. *Matroid theory.* Oxford University Press, Oxford; New York.

PAUL, F., DANILO, V. and FABIO, R. 2007. The myth of non-overlapping channels: interference measurements in IEEE 802.11. *Fourth Annual Conference on Wireless on Demand Network Systems and Services, WONS '07* 1-8.

RAMACHANDRAN, K., BELDING, E., ALMEROTH, K. and BUDDHIKOT, M. 2006. Interference Aware Channel Assignment in Multi-Radio Wireless Mesh Networks. *IEEE INFOCOM*.

RAMACHANDRAN, K., BELDING, E., ALMEROTH, K. and BUDDHIKOT, M. 2006. Interference Aware Channel Assignment in Multi-Radio Wireless Mesh Networks. *IEEE INFOCOM*, 1-12.

RANIWALA, A. and CHIUEH, T.-C. 2005. Architecture and algorithms for an IEEE 802.11-Based Multi-Channel Wireless Mesh Network. *IEEE INFOCOM*, 2223 - 2234

RANIWALA, A., GOPALAN, K. and CHIUEH, T.-C. 2004. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review* 8 (2).

RANIWALA, A., GOPALAN, K. and CHIUEH, T.-C. 2004. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review* 8 (2), 50-65.

TANG, J., XUE, G. and ZHANG, W. 2005. Interference-aware topology control and QoS routing in multi-channel wireless mesh networks. *ACM MobiHoc*, 68-77.

TROPOSNETWORKS. 2005. Tropos Networks MetroMesh Architecture. http://www.tropos.com, White Paper.

VILLAMIZAR, C., CHANDRA, R. and GOVINDAN, R. 1998. BGP Route Flap Damping. *RFC 2439, Standards Track.*

XU, K., GERLA, M. and BAE, S. 2003. Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. *Ad Hoc Networks* 1 (1), 107-123.

XU, S. and SAADAWI, T. 2001. Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks? *IEEE Communications Magazine* 39 (6), 130-137.

ZHAO, B.Y., LING, H., STRIBLING, J., JOSEPH, A.D. and KUBIATOWICZ, J.D. 2003. Exploiting routing redundancy via structured peer-to-peer overlays. *Proceedings of 11th IEEE International Conference on Network Protocols (ICNP'03)* 246-257.

School of Information Technologies, J12
The University of Sydney
NSW 2006 AUSTRALIA
T +61 2 9351 3423
F +61 2 9351 3838
www.it.usyd.edu.au

ISBN 978 1 86487 922 3