Joint Optimal Channel Assignment and Congestion Control for Multi-channel Wireless Mesh Networks

A. Hamed Mohsenian Rad and Vincent W.S. Wong
Department of Electrical and Computer Engineering
The University of British Columbia, Vancouver, Canada
e-mail: {hamed, vincentw}@ece.ubc.ca

Abstract— **The aggregate capacity of wireless mesh networks can be increased by the use of multiple channels. Stationary wireless routers are equipped with multiple network interface cards (NICs). Each NIC is assigned with a distinct frequency channel. In this paper, we formulate the Joint Optimal Channel Assignment and Congestion Control (JOCAC) as a decentralized utility maximization problem with constraints that arise from the interference of the neighboring transmissions. Unlike other previous work, the JOCAC algorithm is able to assign not only the non-overlapping (orthogonal) channels, but also the partially-overlapping channels within the IEEE 802.11 frequency bands. Using 802.11b with 3 non-overlapping channels, simulation results show that our algorithm provides a higher aggregated goodput than the recently proposed load-aware algorithm by 20%. The goodput is further increased by 40% when all the 11 partially-overlapping channels are being used.**

I. Introduction

Wireless mesh networks (WMNs) [1] consist of a number of stationary wireless routers interconnected by wireless links. These wireless routers serve as access points (APs) for wireless mobile devices. Some of them also act as gateways to the Internet via high-speed wired links. Wireless mobile devices first transfer data to the associated wireless router, and these data are then transferred to the Internet (or other networks) via the intermediate wireless routers in a multi-hop manner (see Fig. 1). There have been several implementation studies on wireless mesh networks to provide last-mile broadband Internet access [2] [3]. Some vendors have also begun to offer products in this area [4]–[6].

In our work, we focus on improving the aggregate capacity of the IEEE 802.11a/b/g-based WMNs via the use of multiple channels in each router. In this scenario, stationary wireless routers are assumed to have multiple network interface cards (NICs). Each network interface is assigned to a distinct frequency channel. A router can establish a *link* with a neighboring router when each router has one of its interfaces using the same channel. The number of available channels depends on the frequency band. For example, the IEEE 802.11b/g standards have 11 partially overlapping channels within the 2.4 GHz frequency band, of which 3 channels are non-overlapping. Since the number of channels is limited, some links in the WMNs may be allocated to the same channel. In this case, interference will occur if these links are close to each other. Interference between neighboring links can potentially cause network congestion. For data connections which use TCP (Transmission Control Protocol), if the links become congested, there will be a reduction of the aggregated throughput. Thus, an efficient channel assignment algorithm is crucial to the reduction of interference due to neighboring transmissions using the same channel.

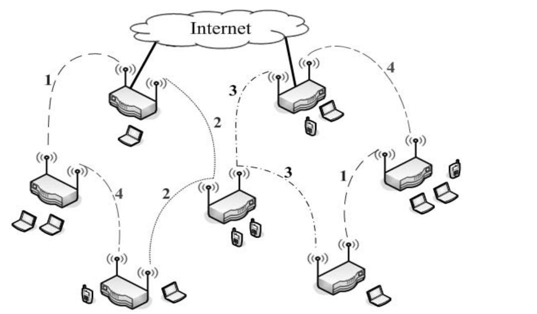

Fig. 1. A multi-channel wireless mesh network with 4 channels.

In this paper, we formulate the Joint Optimal Channel Assignment and Congestion Control (JOCAC) as a decentralized utility maximization problem with constraints that arise from the interference of the neighboring transmissions. The contributions of our work are as follows:

- We present a formulation of the JOCAC problem by taking into account the available number of channels, the number of allocated NICs in each wireless router, the link's congestion price, transmission power, wireless path loss information, and channel frequency response.
- To make efficient use of the available wireless resources, we consider both the non-overlapping and partially overlapping channels in the algorithm. To the best of our knowledge, this is the first paper which considers the partially-overlapping channels in WMNs.
- Simulation results show that our proposed algorithm has a higher aggregated goodput than the recently proposed load-aware algorithm [7]. There is a significant performance gain when all the partially-overlapping channels are being used.

The rest of this paper is organized as follows. The background and related work are described in Section II. The problem formulation is described in Section III. Our proposed algorithm is presented in Section IV. Performance evaluation and comparison are given in Section V. Conclusions and future work are given in Section VI.

II. BACKGROUND AND RELATED WORK

A. Multi-Channel Assignment

The multi-channel assignment algorithms can be divided into two groups. In the first group, each wireless router has a single NIC and channels are selected on a packet-by-packet basis. Some of the multi-channel Medium Access Control (MAC) protocols belong to this group [8], [9]. In the second group, each wireless router has multiple NICs and a distinct channel is allocated to each NIC [7], [10]–[12]. Our proposed algorithm belongs to this group.

An *identical channel assignment* scheme is used in [10], where the 1^{st} channel is assigned to the 1^{st} NIC, 2^{nd} channel to the 2^{nd} NIC, and so on. Although it is simple to implement, the performance is far from optimal. A *centralized* channel assignment and routing algorithm is proposed in [12]. Simulation results show that it provides a higher goodput than the identical channel assignment scheme.

A multi-channel WMN architecture (called *Hyacinth*) is proposed in [7]. The channel assignment problem is divided into two sub-problems. The first one (called *neighbor-to-interface* binding problem) determines through which interface a node should use to communicate with its neighbors. The second one (called *interface-to-channel* binding problem) determines which radio channel an interface should use.

The logical topology of *Hyacinth* has a tree structure [7]. The tree construction mechanism is similar to the IEEE 802.1D spanning tree formation. The gateways are the roots. Each node (i.e., wireless mesh router) uses an *UP-NIC* to connect to its parent and several *DOWN-NICs* to connect to its children. For a given node, the channel of the UP-NIC is assigned by its parent node. As a result, each router is only responsible for channel assignment of the links connected to its DOWN-NICs. *Hyacinth* also uses a *load-aware* algorithm for interface-to-channel binding. It selects the channel which is less used in the neighborhood and requires the nodes to exchange their channel usage information to the interfering neighbors.

In [13], the impact of switching delay (i.e., the time it takes for an interface to switch from one channel to another) is studied. The proposed hybrid interface assignment algorithm in [13] does not require special coordination mechanisms between neighboring nodes. In [11], a joint channel assignment and routing problem is proposed to maximize the bandwidth allocated to each traffic aggregation point, subject to the *fairness constraint*. In [14] [15], the theoretical lower and upper bounds on the multi-channel network capacity are derived.

B. Congestion Control and Network Utility Maximization

Over the past decade, various analytical models have been proposed to study the TCP congestion control in the Internet [16]. One of them, which provides a general framework to analyze different TCP congestion control algorithms, is the idea of *distributed network utility maximization* for elastic traffic sources [17]. This model has also been used to develop different cross-layer congestion control mechanisms (e.g., [18]–[20]) recently.

In the distributed network utility maximization problem formulation, the network is assumed to have S elastic traffic sources and L links. Any given source $s \in S$ transmits its packets with a rate of r_s through a routing path of $L(s) \subset L$. The objective is to maximize the aggregated utility across all sources, subject to the link capacity constraints. The utility is represented by a positive, continuously differentiable, monotone increasing and strictly concave function $U_s(r_s)$. Thus, we have

$$\begin{aligned} &\max_{r \geq 0} \sum_s U_s(r_s) \\ &\text{subject to} \sum_{s: l \in L(s)} r_s \leq c_l \end{aligned} \tag{1}$$

where c_l is the capacity of link l. Note that the function U_s depends on which TCP congestion control algorithm being used. As an example, for TCP Vegas: $U_s(r_s) = \alpha_s d_s \log r_s$, where α_s is the protocol parameter and d_s is the round-trip propagation delay [21]. We will later extend this framework to our channel assignment problem.

III. PROBLEM FORMULATION

In this section, we describe the model formulation of the channel assignment problem. Recall that WMNs consist of a set of stationary wireless routers and some of them also act as gateways to the Internet. We assume that the paths between the routers and the gateways have been pre-determined. For example, the neighbor-to-interface binding mechanism in [7] can be used to determine the paths and the logical topology of the network. In this paper, we focus on the channel assignment on each network interface.

Consider a WMN with N stationary wireless nodes (routers) and L unidirectional links. Each node is equipped with I network interface cards. There are C channels available. We define a *binary channel assignment vector* of $X_{LC \times 1}$ as follows:

$$x_{(l-1)C+c} = \begin{cases} 1, & \text{if } l^{th} \text{ link uses the } c^{th} \text{ channel} \\ 0, & \text{otherwise} \end{cases} \quad \text{for } c = 1, \ldots, C, \quad l = 1, \ldots, L. \tag{2}$$

Since only one frequency channel can be assigned to each given logical link l, among the list of elements $x_{(l-1)C+1}, x_{(l-1)C+2}, \ldots, x_{lC}$, only one of them is equal to 1 and the rest are equal to 0. Thus, we have the following equality constraints:

$$\begin{aligned} &x_{(l-1)C+1} + \cdots + x_{(l-1)C+C} = 1, \ \forall \, l = 1, \ldots, L \\ &\Rightarrow \quad AX = \mathbf{1} \end{aligned} \tag{3}$$

The dimension of matrix A is $L \times LC$.

The second constraint is imposed by the logical topology of the network. The logical network topology is the solution of the neighbor-to-interface binding problem. This constraint requires some links from a given node to use the same frequency channel. That is, if two links u and v from a given node are assigned to use the same NIC, then these two links need to be assigned to the same frequency channel. This can be expressed as:

$$x_{(u-1)C+c} = x_{(v-1)C+c} \ , \quad \forall\, c = 1, \ldots, C \\ \Rightarrow \quad BX = \mathbf{0} \tag{4}$$

For each row in matrix B, two of the entries are equal to 1 and -1, respectively, and all other entries are equal to 0. The dimension of B depends on the number of link pairs that share a common interface.

The vector definition in (2) and the *linear* equality constraints in (3) and (4), together, represent the following *non-empty feasible set*:

$$\Psi = \left\{ X : x \in \{0,1\} \bigcap AX = \mathbf{1} \bigcap BX = \mathbf{0} \right\} \tag{5}$$

Any member of Ψ represents one feasible channel assignment allocation.

Now consider any two arbitrary links i and j, and their associated elements in vector X. We define two $C \times 1$ vectors as follows:

$$X_i = \begin{bmatrix} x_{(i-1)C+1} & x_{(i-1)C+2} & \cdots & x_{iC} \end{bmatrix}^T \\ X_j = \begin{bmatrix} x_{(j-1)C+1} & x_{(j-1)C+2} & \cdots & x_{jC} \end{bmatrix}^T \tag{6}$$

In addition, we define $W_{C\times C}$ as the *channel weighting matrix*. The element $w_{mn} \in [0,1]$ represents the interfering/overlapping portion between channels m and n. W is a symmetric matrix and its diagonal elements are all equal to 1.

If channels m and n are assigned to links i and j respectively, we have

$$X_i^T \ W \ X_j = w_{mn} \tag{7}$$

As an example, suppose the three non-overlapping channels in the IEEE 802.11b band are used for channel assignment (i.e., $C = 3$). W becomes a 3×3 unitary matrix. If two arbitrary links i and j are assigned the same channel, then $X_i^T \ W \ X_j = 1$. Otherwise, the product is equal to zero.

IV. JOINT OPTIMAL CHANNEL ASSIGNMENT AND CONGESTION CONTROL (JOCAC)

A. Theory

In WMNs, the capacity of a virtual (logical) link is not fixed. It is a function of various network parameters, including the transmission powers P, node positions Θ, and the assigned channels X. By assuming the presence of an Additive White Gaussian Noise (AWGN) channel, the virtual link capacity can be expressed as [22]:

$$c_l\left(P, \Theta, X\right) = \frac{1}{T} \log\left(1 + K\, \mathit{SINR}_l\left(P, \Theta, X\right)\right) \tag{8}$$

where T is the symbol period, K is a constant and depends on the modulation scheme being used, and SINR_l is the *Signal to Interference and Noise Ratio* on link l. In wireless mesh networks, nodes (wireless routers) are stationary and their positions are known in advance. The assigned channels X and transmission powers P can be considered as the adjustable variables. However, in this paper we limit our study to fixed transmission powers. Thus, the channel capacity is only a nonlinear function of the assigned channels. Note that the case of single-channel with variable transmission power has already been investigated in [20].

Back to equation (1), we can formulate the utility maximization in WMNs as follows:

$$\max_{X\in\Psi, r\geq 0} \sum_s U_s(r_s) \\ \text{subject to} \sum_{s:l\in L(s)} r_s \leq c_l(X) \tag{9}$$

Based on the Karush-Kuhn-Tucker (KKT) conditions from optimization theory [23], solving problem (9) is equivalent to finding the stationary points of the Lagrangian. Using the *duality* approach [24], the congestion prices (e.g., link delay in TCP Vegas [21], link loss probability in TCP Reno [16]) appear as Lagrange multipliers. The lagrangian is:

$$L(r, \lambda, X) = \sum_s U_s(r_s) - \sum_s \sum_{l\in L(s)} \lambda_l r_s + \sum_l \lambda_l c_l(X) \tag{10}$$

where λ_l is the congestion price on link l. By linearity of the differentiation operator, the problem is decomposed into two separate problems: 1) Congestion control problem, which is solved by the TCP congestion control mechanism. 2) Channel assignment problem. The channel assignment problem is:

$$\max_{X\in\Psi} \sum_l \lambda_l c_l(X) \tag{11}$$

The link capacity c_l in (8) should now be expressed explicitly as a function of X. That is, we need to express *SINR* in terms of X. Since *SINR* is the fraction of signal power to the noise and interference power, we use the concept of *path loss matrix* $\Phi_{N\times N}$ from power control algorithms [25]:

$$\Phi(i,j) = \frac{\text{Received power in node } i}{\text{Transmitted power from node } j} \tag{12}$$

Given two links l and k, and their associated transmitters and receivers (T_l and R_l, T_k and R_k), we define the *link-to-link gain* (g_{lk}) as

$$g_{lk} = \begin{cases} \Phi(R_l, T_l), & l = k \\ \Phi(R_l, T_k), & l \neq k \text{ are activated together} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where l and k cannot be activated simultaneously if there is a shared NIC between T_l and T_k or R_l and R_k.

By using equations (7), (13), and the given transmission power, we can now formulate the SINR_l as follows:

$$\mathit{SINR}_l(X) = \frac{g_{ll}\ p_l}{\sum_{k\neq l\in L} X_k^T W X_l\ p_k g_{lk} + n_l} \tag{14}$$

where p_l is the transmit power of sending node on link l, and n_l is the additive thermal white noise power. We can reasonably assume that the *SINR* is much larger than 1. In this case, the term "1" in (8) can be neglected. Without loss of generality, we assume that the symbol period $T = 1$. The objective function (11) can now be expressed in terms of X:

$$\begin{aligned} \max_{X \in \Psi} \sum_l \lambda_l \log \frac{g_{ll} p_l}{\sum_{k \neq l \in L} X_k^T W X_l p_k g_{lk} + n_l} \\ = \sum_l \lambda_l \log (g_{ll} p_l) \\ - \min_{X \in \Psi} \sum_l \lambda_l \log(\sum_{k \neq l} X_k^T W X_l p_k g_{lk} + n_l) \end{aligned} \quad (15)$$

In (15), the first term is not a function of X. Thus, we only need to minimize the second term.

To avoid service disruption, the channel allocated to each interface will not be changed frequently. In this case, it is more appropriate to use the *average congestion price* $\bar{\lambda}$ instead of the instantaneous congestion price λ. $\bar{\lambda}$ is updated by the following equation:

$$\bar{\lambda_l} \leftarrow (1 - \omega) \bar{\lambda_l} + \omega \lambda_l \quad (16)$$

where ω is the weight and $0 < \omega < 1$. The utility maximization of (15) is then reduced to the following *interference minimization*:

$$\min_{X \in \Psi} \sum_l \bar{\lambda}_l \log \left(\sum_{k \neq l \in L} X_k^T W X_l p_k g_{lk} + n_l \right) \quad (17)$$

Intuitively, if a particular link is congested, its transmission rate can be increased by either increasing the *SINR* or reducing the interference level. This can be achieved by not allocating the same channel used by the congested link to other links within the neighbourhood.

Since the term $\sum_{k \neq l} X_k^T W X_l$ is a *quadratic* function of X, we can call the minimization of (17) as a Sum-Log-Quadratic or simply *Log-Quadratic* method.

Another way to formulate the problem is to remove the log part and to solve the following optimization problem:

$$\min_{X \in \Psi} \sum_l \bar{\lambda}_l \sum_{k \neq l \in L} X_k^T W X_l p_k g_{lk} + n_l \quad (18)$$

We call the problem in (18) as the *Quadratic* method. Although the *Quadratic* method does not follow completely the utility maximization approach of (15), simulation results show that it is less sensitive to transmit power and thermal noise. The performance comparisons between the *Log-Quadratic* and *Quadratic* methods are presented in Section V.

B. Partially Overlapping Channels

The aggregated throughput of the WMNs can further be increased when the partially overlapping channels are being used as well. In this section, we describe how to obtain the corresponding *channel weighting matrix* W. To evaluate the interference levels when the neighboring links use two *partially overlapping* channels, one needs to know which channel filter is being used. In this paper, we assume the use of the *Raised Cosine Filter* [22]. One of the parameters to identify the frequency response of this filter is the *roll-off factor* δ. Fig. 2 shows the response of the IEEE 802.11b channels with δ equals to 1 and 0.25, respectively.

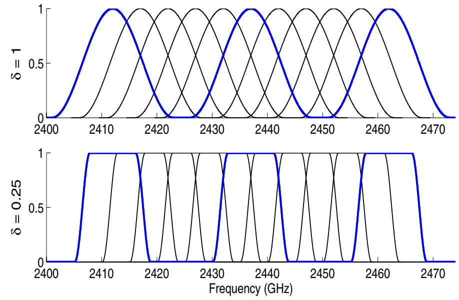

Fig. 2. The available 11 partially overlapping channels in IEEE 802.11b.

For any two arbitrary channels m and n, the value of w_{mn} in W is given by:

$$w_{mn} = w_{nm} = \frac{A_o}{A_o + A_{no}} \quad (19)$$

where A_o and A_{no} are the overlapping and non-overlapping areas between the *power spectral density* (PSD) of channels m and n, respectively. The PSD can be derived from the channels' frequency responses. The above calculation can be performed off-line and later be used as a constant matrix W in the JOCAC algorithm implementation.

C. Implementation

The JOCAC algorithm can be implemented either in a centralized or distributed manner. For centralized implementation, one of the gateway nodes can solve (17) or (18) and announce the selected channels X to the wireless nodes. For distributed implementation, each node n is responsible for assigning the optimal channels to some links $L_n \subset L$. Each node n needs to periodically exchange its individual channel usage $X_l, l \in L_n$, as well as its collected data $\lambda_l, l \in L_n$ with all other nodes. The local distributed implementation of (17) is given in (20). Problem (18) can be locally implemented in a similar way.

$$\begin{aligned} \min_{X_{i,\, i \in L_n}} \sum_l \bar{\lambda}_l \log \left(\sum_{k \neq l \in L} X_k^T W X_l p_k g_{lk} + n_l \right) \\ \text{subject to} \quad X \in \Psi \\ X_{j,\, j \in L,\, j \notin L_n} \end{aligned} \quad (20)$$

The *exhaustive search* is used to obtain either the optimal solution of (17) or the partially optimal solution of (20).

V. Performance Comparison

In this section, we compare the performance between our proposed JOCAC algorithm and the load-aware algorithm [7]. For the JOCAC algorithm, we use the notations LQ-JOCAC and Q-JOCAC to denote the *Log-Quadratic* and *Quadratic* methods described in Section IV. We also present the results when all 11 partially-overlapping frequency channels in the IEEE 802.11b 2.4 GHz band are being used.

The comparisons are conducted by using the *ns-2* simulator. Besides using the patch from [26], we modified several modules in ns-2 to improve the accuracy of the model. First of all, *ns-2* does not keep track of the *total* interference power; however the actual interference is *cumulative*. We modified the *ns-2* code so that the *SINR* is calculated by using the power from all signals currently on the channel. Secondly, the Auto-Rate Fallback (ARF) protocol, which is widely implemented in commercial products, is not implemented in *ns-2*. By using the patch of ARF from [27], we further increased the accuracy by adaptively changing the *capture threshold* based on the selected transmission rate. That is, the higher the transmission rate, the higher the required *SINR* to capture the packet. The capture thresholds are selected by using the *BER-SINR* curves of the IEEE 802.11 a/g PHY mode [28] with target BER of 10^{-6} and packet size of 1024 bytes. Finally, *ns-2* does not support partially overlapping channels. We modified it so that the wireless interfaces capture the packets from *all* the partially overlapping channels, unless the packet's interference portion is less than the *carrier sensing threshold*.

A. Topologies, Traffic Patterns, and Evaluation Metric

In the simulation model, the size of the network is $1000\text{m} \times 500\text{m}$. Ten different topologies are randomly generated. Each topology consists of 2 gateways and 15 randomly scattered wireless mesh nodes. Since we intend to compare our JOCAC algorithm with the load-aware algorithm [7], we use the *Hyacinth* architecture to create the logical network topology. That is, the logical topology is obtained via the neighbor-to-interface binding mechanism in [7]. One of the logical network topologies we used in our simulation is shown in Fig. 3.

Each wireless node has two NICs: one as DOWN-NIC and one as UP-NIC. The communication range and the carrier sensing/interference range are 250m and 450m, respectively.

In each topology, five different randomly generated traffic patterns are used. Each traffic pattern consists of 30 TCP Vegas flows. 15 of them are always-on flows and the other 15 are randomly-on flows. Any given node is the source of exactly one always-on flow; the destination is selected randomly. As a result, there is always some load on each link. The randomly-on flows have arbitrarily selected sources and destinations; these flows introduce additional traffic on the links. The lifetime of each randomly-on flow follows a uniform distribution between 0 and 300 sec. For TCP Vegas, we choose $\alpha = 3$ and $\beta = 9$. It leads to a larger TCP congestion window size and consequently a higher load on the network.

Since the original load-aware algorithm [7] does not support the elastic TCP traffic, we modified the original load-aware algorithm and approximated the TCP traffic load [16] as:

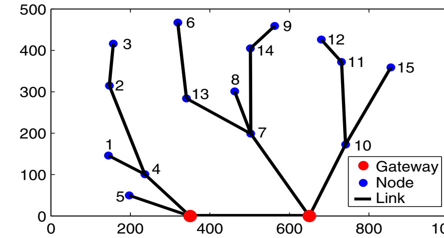

Fig. 3. One of the simulation topologies (Topology # 1).

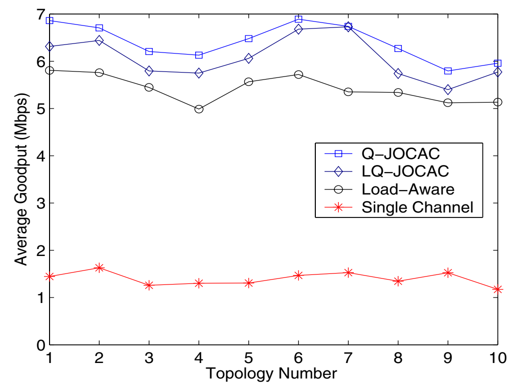

Fig. 4. Average goodput by using 3 non-overlapping channels.

$$\text{TCP Traffic Load} = \frac{cwnd}{rtt} \tag{21}$$

where $cwnd$ is the TCP congestion window size and rtt is the average packet's round-trip time. The above rate is exchanged between each source and other nodes in the neighborhood.

The *goodput* of a given TCP connection is the total correctly received bytes at the destination divided by the total simulation time. The aggregated goodput of all 30 TCP flows is the *network's cross-section goodput*. For a given topology, the *average goodput* is determined by averaging the network's cross-section goodput from five different traffic patterns. In all simulation runs, the simulation time is 300 sec. The channel assignment algorithms are invoked every 60 sec.

B. Simulation Results

Fig. 4 shows the results of the average goodput when 3 non-overlapping channels in the IEEE 802.11b 2.4 GHz band are being used. Results show that the load-aware, LQ-JOCAC and Q-JOCAC algorithms have a higher average goodput than the single-channel case by a factor of 3.8, 4.3, and 4.6, respectively. In addition, LQ-JOCAC and Q-JOCAC algorithms have a higher average goodput than the load-aware algorithm by 13.2% and 21.1%, respectively.

The Q-JOCAC algorithm has a slightly higher average goodput than the LQ-JOCAC since the *Log-Quadratic* method may occasionally encounter the *rapid switching problem*. That is, the *Log-Quadratic* method is more sensitive to the changes of the interference level. In (17), when the interference level changes from zero to a small value, it may incur a relatively high cost value which results in a change of channels. Results

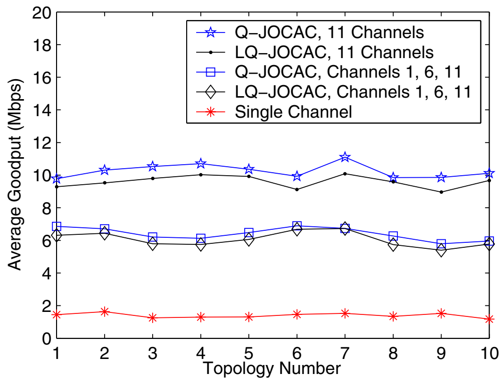

Fig. 5. Performance comparison between using the 3 non-overlapping channels and all the partially-overlapping channels.

show that at each step, the LQ-JOCAC algorithm works in favor of the congested links by restricting the neighboring links from using the same channels. However, it causes congestion in the neighboring links at a later time. On the other hand, results show that the *Quadratic* cost function in (18) leads to a smooth change in the interference levels, and consequently provides more stable channel assignments.

Fig. 5 shows the performance gain between using only 3 non-overlapping channels and using all the available 11 partially-overlapping channels in the IEEE 802.11b 2.4 GHz band. For both LQ-JOCAC and Q-JOCAC algorithms, the channel's roll-off factor δ is equal to 0.25. Results show that when all the partially-overlapping channels are being used, both LQ-JOCAC and Q-JOCAC algorithms have a higher average goodput than the single-channel case by a factor of 6.9 and 7.3, respectively. In addition, for Q-JOCAC algorithm, its average goodput is further increased by 40% when all the partially-overlapping channels are being used.

VI. Conclusions

In this paper, we proposed a Joint Optimal Channel Assignment and Congestion Control (JOCAC) algorithm for multi-channel wireless mesh networks. The JOCAC algorithm allocates channels to control the interference on each link regarding the link's average congestion price. One of the distinct advantages of this algorithm is the ability to assign not only the non-overlapping channels, but also the partially-overlapping channels. This allows the IEEE 802.11 frequency band to be fully utilized. Using the 802.11b frequency band as an example, simulation results show that in the non-overlapping case, LQ-JOCAC and Q-JOCAC algorithms have a higher average goodput than the load-aware algorithm [7] by 13.2% and 21.1%, respectively. For Q-JOCAC algorithm, its average goodput is further increased by 40% when all the partially-overlapping channels are being used.

For future work, we plan to extend the algorithm by including routing as a cross-layer design framework. We shall also study the stability conditions and the convergence rate of the JOCAC algorithm.

Acknowledgment

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant number STPGP 269872-03.

References

[1] I. Akyildiz, X. Wang, and X. W. Wang, "Wireless mesh networks: A survey," *Elsevier Journal on Computer Networks*, vol. 47, pp. 445–487, March 2005.
[2] "MIT Roofnet," http://pdos.csail.mit.edu/roofnet/doku.php.
[3] "Microsoft Mesh Networking," http://research.microsoft.com/mesh/.
[4] "Tropos Networks," http://www.tropos.com.
[5] "Meshdynamics Inc." http://www.meshdynamics.com.
[6] "BelAir Networks," http://www.belairnetworks.com.
[7] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE Infocom*, Miami, Florida, March 2005.
[8] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. ACM MobiHoc*, Tokyo, Japan, May 2004.
[9] A. Tzamaloukas and J. Garcia-Luna-Aceves, "A receiver-initiated collision-avoidance protocol for multi-channel networks," in *Proc. IEEE Infocom*, Anchorage, Alaska, April 2001.
[10] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. ACM MobiCom*, Philadelphia, Pennsylvania, Sept. 2004.
[11] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. ACM MobiCom*, Cologne, Germany, Sept. 2005.
[12] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM Mobile Computing and Communications Review*, vol. 8, pp. 50–65, April 2004.
[13] P. Kyasanur and N. Vaidya, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE WCNC*, New Orleans, LA, 2005.
[14] ——, "Capacity of multi-channel wireless networks: Impact of number of channels and interfaces," in *Proc. ACM MobiCom*, Cologne, Germany, Sept. 2005.
[15] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. ACM MobiCom*, Cologne, Germany, Sept. 2005.
[16] S. Low, F. Paganini, and J. Doyle, "Internet congestion control," *IEEE Control Systems Magazine*, pp. 28–43, Feb. 2002.
[17] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecommunication*, vol. 8, pp. 33–37, 1997.
[18] L. Chen, S. H. Low, and J. Doyle, "Joint congestion control and media access control design for ad hoc wireless networks," in *Proc. IEEE Infocom*, Miami, Florida, March 2005.
[19] J. Wang, L. Li, S. H. Low, and J. Doyle, "Cross-layer optimization in TCP/IP networks," *IEEE/ACM Trans. on Networking*, vol. 13, pp. 582 – 595, June 2005.
[20] M. Chiang, "To layer or not to layer: Balancing transport and physical layers in wireless multihop networks," in *Proc. IEEE Infocom*, Hong Kong, China, March 2004.
[21] S. H. Low, L. Peterson, and L. Wang, "Understanding Vegas: A duality model," *Journal of the ACM*, vol. 49, pp. 207–235, March 2002.
[22] J. Proakis, *Digital Communications*. McGraw-Hill, 2000.
[23] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Science, 2004.
[24] S. Low and D. Lapsley, "Optimization flow control, I: Basic algorithms and convergence," *IEEE/ACM Trans. Networking*, vol. 7, pp. 861–874, Dec. 1999.
[25] J. Zender, "Performance of optimal power control in cellular radio systems," *IEEE Trans. Vehicular Tech*, vol. 12, Feb. 1992.
[26] "ns2 patch," www.ecsl.cs.sunysb.edu/multichannel/code/ns-2.1b9a-multi-nic-patch, released 2004.
[27] "ns2 patch," www.winlab.rutgers.edu/∼zhibinwu/html/multirate_ns_mac.htm, released 2003.
[28] D. Qiao, S. Choi, and K. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless LANs," *IEEE Trans. on Mobile Computing*, vol. 1, pp. 278–292, Oct.-Dec. 2002.