Channel Assignment and Routing in Multi-radio Wireless Mesh Networks, 2005; Alicherry, Bhatia, Li

Mansoor Alicherry, Bell Labs, Alcatel-Lucent, www1.bell-labs.com/user/mansoor/
Randeep Bhatia, Bell Labs, Alcatel-Lucent, www.bell-labs.com/user/randeep/
Li (Erran) Li, Bell Labs, Alcatel-Lucent, www.bell-labs.com/user/erranlli/

entry editor: Xiang-Yang Li

INDEX TERMS: Channel assignment, Approximation algorithm, mathematical programming, graph theory, wireless mesh networks
SYNONYMS: graph coloring, ad-hoc networks

1 PROBLEM DEFINITION

One of the major problem facing wireless networks is the capacity reduction due to interference among multiple simultaneous transmissions. In wireless mesh networks providing mesh routers with multiple-radios can greatly alleviate this problem. With multiple-radios, nodes can transmit and receive simultaneously or can transmit on multiple channels simultaneously. However, due to the limited number of channels available the interference cannot be completely eliminated and in addition careful channel assignment must be done to mitigate the effects of interference. Channel assignment and routing are inter-dependent. This is because channel assignments have an impact on link bandwidths and the extent to which link transmissions interfere. This clearly impacts the routing used to satisfy traffic demands. In the same way traffic routing determines the traffic flows for each link which certainly affects channel assignments. Channel assignments need to be done in a way such that the communication requirements for the links can be met. Thus, the problem of throughput maximization of wireless mesh networks must be solved through channel assignment, routing and scheduling.

Formally, given a wireless mesh backbone network modeled as a graph (V, E). The node $t \in V$ represents the wired network. An edge $e = (u, v)$ exists in E iff u and v are within *communication range* R_T. The set $V_G \subseteq V$ represents the set of gateway nodes. The system has a total of K channels. Each node $u \in V$ has $I(u)$ network interface cards, and has an aggregated demand $l(u)$ from its associated users. For each edge e the set $I(e) \subset E$ denotes the set of edges that it interferes with. A pair of nodes that use the same channel and are within *interference range* $R_I x$. may interfere with each other's communication, even if they cannot directly communicate. Node pairs using different channels can transmit packets simultaneously without interference. The problem is to maximize λ where at least $\lambda l(u)$ amount of throughput can be routed from each node u to the Internet (represented by a node t). The $\lambda l(u)$ throughput for each node u is achieved by computing (1) a network flow that associates with each edge $e = (u, v)$ values $f(e(i)), 1 \leq i \leq K$ where $f(e(i))$ is the rate at which traffic is transmitted by node u for node v on channel i; (2) a feasible channel assignment $F(u)$ ($F(u)$ is an ordered set where the $i-th$ interface of u operates on the $i-th$ channel in $F(u)$) such that, whenever $f(e(i)) > 0$, $i \in F(u) \cap F(v)$; In this case edge e is denoted to use channel i (3) a *feasible* schedule S that decides the set of edge channel pair (e, i) (edge e using channel i) scheduled at time slot τ, for $\tau = 1, 2, \ldots, T$ where T is the period of the schedule. A schedule is feasible if the edges of no two edge pairs $(e_1, i), (e_2, i)$ scheduled in the same time slot for a common channel i interfere with each other ($e_1 \notin I(e_2)$ and $e_2 \notin I(e_1)$). Thus, a feasible schedule is also referred to as an interference free edge schedule. An indicator variable $X_{e,i,\tau}, e \in E, i \in F(e), \tau \geq 1$ is used. It is assigned 1 if and only if link e is active in slot τ on channel i. Note that $\frac{1}{T}\sum_{1\leq\tau\leq T} X_{e,i,\tau}c(e) = f(e(i))$. This is because communication at rate $c(e)$ happens in every slot that link e is active on channel i and since $f(e(i))$ is the average rate attained on link e for channel i. This implies $\frac{1}{T}\sum_{1\leq\tau\leq T} X_{e,i,\tau} = \frac{f(e(i))}{c(e)}$.

2 Joint routing, channel assignment and link scheduling algorithm

Even the interference free edge scheduling sub-problem given the edge flows is NP-hard [5]. An approximation algorithm called RCL for the joint routing, channel assignment and link scheduling problem has been developed. The algorithm performs the following steps in the given order:

1. **Solve LP:** First optimally solve a LP relaxation of the problem. This results in a flow on the flow graph along with a not necessarily feasible channel assignment for the node radios. Specifically, a node may be assigned more channels than the number of its radios. However, this channel assignment is "optimal" in terms of ensuring that the interference for each channel is minimum. This step also yields a lower bound on the λ value which is used in establishing the worst case performance guarantee of the overall algorithm.

2. **Channel Assignment:** This step presents a channel assignment algorithm which is used to adjust the flow on the flow graph (routing changes) to ensure a feasible channel assignment. This flow adjustment also strives to keep the increase in interference for each channel to a minimum.

3. **Interference Free Link Scheduling:** This step obtains an interference free link schedule for the edge flows corresponding to the flow on the flow graph.

Each of these steps is described in the following subsections.

2.1 A Linear Programming based Routing Algorithm

A Linear Program (LP1) to find a flow that maximizes λ. is given below:

$$\max \lambda \tag{1}$$

Subject to

$$\lambda l(v) + \sum_{e=(u,v)\in E} \sum_{i=1}^{K} f(e(i)) = \sum_{e=(v,u)\in E} \sum_{i=1}^{K} f(e(i)), \forall v \in V - V_G \tag{2}$$

$$f(e(i)) \leq c(e), \ \forall e \in E \tag{3}$$

$$\sum_{1\leq i\leq K} \left(\sum_{e=(u,v)\in E} \frac{f(e(i))}{c(e)} + \sum_{e=(v,u)\in E} \frac{f(e(i))}{c(e)} \right) \leq I(v), \ v \in V \tag{4}$$

$$\frac{f(e(i))}{c(e)} + \sum_{e'\in I(e)} \frac{f(e'(i))}{c(e')} \leq c(q), \forall e \in E, \ 1 \leq i \leq K \tag{5}$$

The first two constraints are *flow constraints*. The first one is the flow conservation constraint; the second one ensures no link capacity is violated. The third constraint is the *node radio constraints*. Recall that a IWMN node $v \in V$ has $I(v)$ radios and hence can be assigned at most $I(v)$ channels from $1 \leq i \leq K$. One way to model this constraint is to observe that due to interference constraints v can be involved in at most $I(v)$ simultaneous communications (with different one hop neighbors). In other word this constraint follows from $\sum_{1\leq i\leq K}\sum_{e=(u,v)\in E} X_{e,i,\tau} + \sum_{1\leq i\leq K}\sum_{e=(v,u)\in E} X_{e,i,\tau} \leq I(v)$. The fourth constraint is the *link congestion constraints* which is discussed in detail in Section 2.3. Note that all the constraints listed above are necessary conditions for any feasible solution. However, these constraints are not necessarily sufficient. Hence if a solution is found that satisfies these constraints it may not be a feasible solution. The approach is to start with a "good" but not necessarily feasible solution that satisfies all of these constraints and use it to construct a feasible solution without impacting the quality of the solution.

A solution to this LP can be viewed as a flow on a *flow graph* $H = (V, E^H)$ where $E^H = \{e(i)|\forall e \in E, 1 \leq i \leq K\}$. Although the optimal solution to this LP yields the best possible λ (say λ^*) from a practical point of view some more improvements may be possible:

- The flow may have directed cycles. This may be the case since the LP does not try to minimize the amount of interference directly. By removing the flow on the directed cycle (equal amount off each edge) flow conservation is maintained and in addition since there are fewer transmissions the amount of interference is reduced.
- Flow may be going on long path when shorter paths are available. Note that longer paths imply more link transmissions. In this case many times by moving the flow to shorter paths, system interference may be reduced.

The above arguments suggests that it would be practical to find among all solutions that attain the optimal λ value of λ^* the one for which the total value of the following quantity is minimized:

$$\sum_{1\leq i\leq K}\sum_{e=(v,u)\in E}\frac{f(e(i))}{c(e)}.$$

The LP is then re-solved with this objective function and with λ fixed at λ^*.

2.2 Channel Assignment

The solution to the LP (1) is a set of flow values $f(e(i))$ for edge e and channel i that maximize the value λ. Let λ^* denote the optimal value of λ. The flow $f(e(i))$ implies a channel assignment where the two end nodes of edge e are both assigned channel i if and only if $f(e(i)) > 0$. Note that for the flow $f(e(i))$ the implied channel assignment may not be feasible (it may require more than $I(v)$ channels at node v). The channel assignment algorithm transforms the given flow to fix this infeasibility. Below only a sketch of the algorithm is given. More details can be found in [1].

First observe that in an idle scenario, where all nodes v have the same number of interfaces I (i.e. $I = I(v)$) and where the number of available channels K is also I, the channel assignment implied by the LP (1) is feasible. This is because even the trivial channel assignment where all nodes are assigned all the channels 1 to I is feasible. The main idea behind the algorithm is to first transform the LP (1) solution to a new flow in which every edge e has flow $f(e(i)) > 0$ only for the channels $1 \leq i \leq I$. The basic operation that the algorithm uses for this is to equally distribute, for every edge e, the flow $f(e(i))$, for $I < i \leq K$ to the edges $e(j)$, for $1 \leq i \leq I$. This ensures that all $f(e(i)) = 0$, for $I < i \leq K$ after the operation. This operation is called the Phase I of the Algorithm. Note that the Phase I operation does not violate the flow conservation constraints or the node radio constraints (5) in the LP (1). It can be shown that in the resulting solution the flow $f(e(i))$ may exceed the capacity of edge e by at most a factor $\phi = \frac{K}{I}$. This is called the "inflation factor" of Phase I. Likewise in the new flow, the *link congestion constraints* (5) may also be violated for edge e and channel i by no more than the inflation factor ϕ. In other words in the resulting flow

$$\frac{f(e(i))}{c(e)} + \sum_{e'\in I(e)}\frac{f(e'(i))}{c(e')} \leq \phi c(q).$$

This implies that if the new flow is scaled by a fraction $1/\phi$ than it is feasible for the LP (1). Note that the implied channel assignment (assign channels 1 to I to every node) is also feasible. Thus the above algorithm finds a feasible channel assignment with a λ value of at least λ^*/ϕ.

One shortcoming of the channel assignment algorithm (Phase I) described so far is that it only uses I of the K available channels. By using more channels the interference may be further reduced thus allowing for more flow to be pushed in the system. The channel assignment algorithm uses an additional heuristic for this improvement. This is called Phase II of the algorithm.

Now define an operation called "channel switch operation". Let A be a maximal connected component (the vertices in A are not connected to vertices outside A) in the graph formed by the edges e for a given channel i for which $f(e(i)) > 0$. The main observation to use is that for a given channel j, the operation of completely moving flow $f(e(i))$ to flow $f(e(j))$ for every edge e in A, does not impact the feasibility of the implied channel assignment. This is because there is no increase in the number of channels assigned per node after the flow transformation: the end nodes of edges e in A which were earlier assigned channel i are now assigned channel j instead. Thus, the transformation is equivalent to switching the channel assignment of nodes in A so that channel i is discarded and channel j is gained if not already assigned.

The Phase II heuristic attempts to re-transform the unscaled Phase I flows $f(e(i))$ so that there are multiple connected components in the graphs $G(e,i)$ formed by the edges e for each channel $1 \leq i \leq I$. This re-transformation is done so that the LP constraints are kept satisfied with an inflation factor of at most ϕ, as is the case for the unscaled flow after the Phase I of the algorithm.

Next in Phase III of the algorithm the connected components within each graph $G(e,i)$ are grouped such that there are as close to K (but no more than) groups overall and such that the maximum interference within each group is minimized. Next the nodes within the l_{th} group are assigned channel l, by using the "channel switch operation" to do the corresponding flow transformation. It can be shown that the channel assignment implied by the flow in Phase III is feasible. In addition the underlying flows $f(e(i))$ satisfy the LP (1) constraints with an inflation factor of at most $\phi = \frac{K}{I}$.

Next the algorithm scales the flow by the largest possible fraction (at least $1/\phi$) such that the resulting flow is a feasible solution to the LP (1) and also implies a feasible channel assignment. solution to the channel assignment. Thus the overall algorithm finds a feasible channel assignment (by not necessarily restricting to channels 1 to I only) with a λ value of at least λ^*/ϕ.

2.3 Link Flow Scheduling

The results in this section are obtained by extending those of [4] for the single channel case and for the Protocol Model of interference [2]. Recall that the time slotted schedule S is assumed to be periodic (with period T) where the indicator variable $X_{e,i,\tau}, e \in E, i \in F(e), \tau \geq 1$ is 1 if and only if link e is active in slot τ on channel i and i is a channel in common among the set of channels assigned to the end-nodes of edge e.

Directly applying the result (Claim 2) in [4] it follows that a necessary condition for interference free link scheduling is that for every $e \in E, i \in F(e), \tau \geq 1$: $X_{e,i,\tau} + \sum_{e' \in I(e)} X_{e',i,\tau} \leq c(q)$. Here $c(q)$ is a constant that only depends on the interference model. In the interference model this constant is a function of the fixed value q, the ratio of the interference range R_I to the transmission range R_T, and an intuition for its derivation for a particular value $q = 2$ is given below.

Lemma 1. *$c(q) = 8$ for $q = 2$.*

Proof: Recall that an edge $e' \in I(e)$ if there exist two nodes $x, y \in V$ which are at most $2R_T$ apart and such that edge e is incident on node x and edge e' is incident on node y. Let $e = (u, v)$. Note that u and v are at most R_T apart. Consider the region C formed by the union of two circles C_u and C_v of radius $2R_T$ each, centered at node u and node v respectively. Then $e' = (u', v') \in I(e)$ if an only if at least one of the two nodes u', v' is in C; Denote such a node by $C(e')$. Given two edges $e_1, e_2 \in I(e)$ that do not interfere with each other it must be the case that the nodes $C(e_1)$ and $C(e_2)$ are at least $2R_T$ apart. Thus an upper bound on how many edges in $I(e)$ do not pair-wise interfere with each other can be obtained by computing how may nodes can be put in C that are pair-wise at least $2R_T$ apart. It can be shown [1] that this number is at most 8. Thus in schedule S in a given slot only one of the two possibilities exist: either edge e is scheduled or an "independent" set of edges in $I(e)$ of size at most 8 is scheduled implying the claimed bound. ■

A necessary condition:(*Link Congestion Constraint*) Recall that $\frac{1}{T}\sum_{1\le\tau\le T} X_{e,i,\tau} = \frac{f(e(i))}{c(e)}$. Thus: Any valid "interference free" edge flows must satisfy for every link e and every channel i the Link Congestion Constraint:

$$\frac{f(e(i))}{c(e)} + \sum_{e'\in I(e)} \frac{f(e'(i))}{c(e')} \le c(q). \tag{6}$$

A matching sufficient condition can also established [1].

A sufficient condition:(*Link Congestion Constraint*) If the edge flows satisfy for every link e and every channel i the following *Link Schedulability Constraint* than an interference free edge communication schedule can be found using an algorithm given in [1].

$$\frac{f(e(i))}{c(e)} + \sum_{e'\in I(e)} \frac{f(e'(i))}{c(e')} \le 1. \tag{7}$$

The above implies that if a flow $f(e(i))$ satisfies the *Link Congestion Constraint* then by scaling the flow by a fraction $1/c(q)$ it can be scheduled free of interference.

3 KEY RESULTS

Theorem 2. *The RCL algorithm is a* $\frac{Kc(q)}{I}$ *approximation algorithm for the Joint Routing and Channel Assignment with Interference free Edge Scheduling problem.*

PROOF: Note that the flow $f(e(i))$ returned by the channel assignment algorithm in Section 2.2 satisfies the *Link Congestion Constraint*. Thus, from the result of Section 2.3 it follows that by scaling the flow by an additional factor of $1/c(q)$ the flow can be realized by an interference free link schedule. This implies a feasible solution to the joint routing, channel assignment and scheduling problem with a λ value of at least $\frac{\lambda^*}{\phi c(q)}$. Thus the RCL algorithm is a $\phi c(q) = \frac{Kc(q)}{I}$ approximation algorithm. ■

4 APPLICATIONS

Infrastructure mesh networks are increasingly been deployed for commercial use and law enforcement. These deployment settings place stringent requirements on the performance of the underlying IWMNs. Bandwidth guarantee is one of the most important requirements of applications in these settings. For these IWMNs, topology change is infrequent and the variability of aggregate traffic demand from each mesh router (client traffic aggregation point) is small. These characteristics admit periodic optimization of the network which may be done by a system management software based on traffic demand estimation. This work can be directly applied to IWMNs. It can also be used as a benchmark to compare against heuristic algorithms in multi-hop wireless networks.

5 OPEN PROBLEMS

For future work, it will be interesting to investigate the problem when routing solutions can be enforced by changing link weights of a distributed routing protocol such as OSPF. Also, can the worst case bounds of the algorithm be improved (e.g. a constant factor independent of K and I)?

6 EXPERIMENTAL RESULTS

Please refer to [1].

7 DATA SETS

Please refer to [1].

8 URL to CODE

None is reported.

9 CROSS REFERENCES

Graph coloring, 00474
Broadcast scheduling, 00462

10 RECOMMENDED READING

[1] M. Alicherry and R. Bhatia and L. E. Li. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Proc. ACM MOBICOM*, pp. 58–72, 2005.

[2] P. Gupta and P. R. Kumar. The Capacity of Wireless Networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, Mar. 2000.

[3] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *Proc. ACM MOBICOM*, pp. 66–80, 2003.

[4] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan. Algorithmic aspects of capacity in wireless networks. In *Proc. ACM SIGMETRICS*, pp. 133–144, 2005.

[5] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan. End-to-end packet-scheduling in wireless ad-hoc networks. In *Proc. ACM-SIAM symposium on Discrete algorithms*, pp. 1021–1030, 2004.

[6] P. Kyasanur and N. Vaidya. Capacity of multi-channel wireless networks: Impact of number of channels and interfaces. In *Proc. ACM MOBICOM*, 2005.