This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

Scheduling Multiple Partially Overlapped Channels in Wireless Mesh Networks

Haiping Liu   Hua Yu   Xin Liu   Chen-Nee Chuah   Prasant Mohapatra
University of California, Davis
Email: { hpliu, huayu, xinliu, chuah, pmohapatra }@ucdavis.edu

Abstract— **In this paper, we explore the use of partially overlapped channels in wireless mesh networks that consist of multiple 802.11-based access points. We propose novel channel allocation and link scheduling algorithms in the MAC layer to enhance network performance. Due to different traffic characteristics in multi-hop WMNs compared to those in one-hop 802.11 networks, we perform our optimization based on end-to-end flow requirement, instead of the sum of link capacity. In addition, we discuss other factors affecting the performance of POC, including topology, node density, and distribution.**

I. Introduction

In recent years, wireless mesh networks (WMNs) have become an increasingly popular option for providing ubiquitous network access to users [2]. Currently, the most common WMNs are infrastructure-based multi-hop wireless networks, consisting of mesh routers and client nodes. Such backhaul network architecture is reliable, scalable, and easy to deploy. However, the capacity in WMNs is limited. Currently in 802.11-based WMNS, two nodes can only communicate when they are on fully-overlapped channels (FOC), i.e. the same channel. The number of simultaneous transmissions is limited by interference, and the system capacity degrades due to the multi-hop nature of WMNs [7]. Thus, efficient bandwidth utilization is important and recent research focuses on increasing the capacity of WMNs while maintaining connectivity.

IEEE 802.11 is a widely used radio technology for WMNs. The most popular variants, 802.11b and 802.11g, operate in the ISM 2.4 GHz band that has 11 available channels in the USA, of which three are orthogonal. Since each channel is 22 MHz wide and only 5 MHz separates the center frequencies of neighboring channels, a signal on one channel will interfere with several adjacent ones.

Recently, there has been a significant amount of research in the area of WMNs to enhance system capacity by switching channels [4], [8], [15], [16] and/or using multiple radios [1], [3], [6], [14]. In SSCH [4], nodes randomly switch channels such that the neighboring nodes meet periodically at a common channel to communicate. Both Multi-NIC [14] and MUP [1] utilize multiple radios to improve capacity. Multi-NIC focuses on channel assignment, but uses a simplified interference model. MUP advocates unifying multiple radios and abstracting their use at higher layers. These works only consider non-overlapping channels.

Recently, there are some studies on the mechanism of partially overlapped channels (POC) [9]–[12], which permits sender and receiver, or adjacent sender-receiver pair, to use partially overlapped channels to communicate. In [9], [10], Mishra et.al. propose this new idea and measure the receiving power among different channels. In [11], they further analyze the improvement of POC based on CSMA/CA in one-hop 802.11 networks, and adopt existing algorithms for POC channel allocation. In [12], A. Rad and V. Wong also propose a decentralized algorithm for *POC* within the IEEE 802.11 frequency bands.

Compared to their studies based on random access, this paper considers controlled access and proposes a joint channel allocation and link scheduling algorithms for POC in WMN.

Our main contributions include:

1) Given the topology of a WMN, we propose heuristic algorithms to allocate multiple channels to the nodes and schedule the links into different time slots;
2) We consider end-to-end flow requirements in WMN instead of the sum of link capacities;
3) We discuss other factors influencing the capacity improvement by POC, such as topology, node density and, node distribution.

II. Modeling and Formulation

A. PHY Layer Modeling

Four major factors dominate point-to-point transmission performance: signal attenuation due to distance, frequency overlapping, interference from other simultaneous transmissions, and coding. We construct a model to evaluate the POC performance given these factors in the following.

The first factor prevails in all wireless communications. We adopt the popular scattering model. Let E_s and E_r be sending and receiving power respectively, and let d be the distance between sender and receiver. Then

$$E_r = k * d^{-\alpha} * E_s, \qquad (1)$$

where k is a constant related to the gains of sender and receiver antennas, and α is the scattering parameter.

In the POC system, the sender and the receiver may share partially overlapped channel as shown in Fig. 1. Let the power density function of the sender be $P_r(f)$ and channel response be $H(f)$. The channel width is $2\frac{1}{T_c}$, and the bandwidth shift is ΔF. The actual signal power at the receiver is

$$E_r = k * d^{-\alpha} \int_{-\frac{1}{T_c}+\Delta F}^{\frac{1}{T_c}+\Delta F} P_r(f)\, H^2(f) \mathrm{d}f. \qquad (2)$$

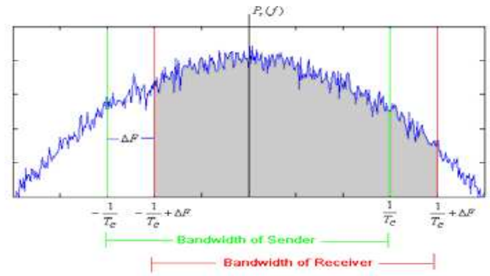


Fig. 1. The Bandwidth of Sender and Receiver

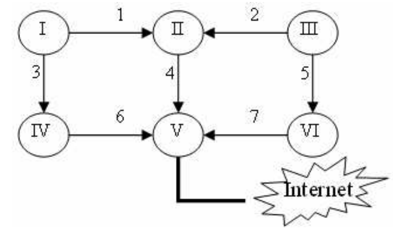


Fig. 2. The topology of an example WMN

The most attractive characteristic of POC is the possibility of multiple transmissions at the same time. On the other hand, one current transmission may suffer interference from other simultaneous transmissions in addition to the channel noise. The interference power E_i of each of the simultaneous transmissions can also be derived from (2), where $P_r(f)$ and ΔF are from the interfering nodes. We employ the SNR model and let the channel noise power be N_0, then

$$SNR = \frac{Er}{N_0 + \sum E_i} \ . \tag{3}$$

From the SNR, we can derive the bit-error-rate P_{BER} according to different modulation schemes.

In current wireless communication, block-coding is the most popular scheme. Let $LenP$ be the packet length, and m be the minimally required number of correctly received bits to ensure successful decoding. The probability of successfully receiving a packet P_{sp} is

$$P_{sp} = \sum_{k=m}^{LenP} \binom{LenP}{k} (P_{BER})^{LenP-k} (1 - P_{BER})^k. \tag{4}$$

Consequently the link capacity B is:

$$B = \frac{Data * P_{sp}}{T} \ . \tag{5}$$

where $Data$ is the length of payload in one packet, and T is the duration of a transmission. We use this model for performance evaluation in the rest of the paper.

B. MAC Layer Formulation

In this part, we propose the MAC layer formulation to optimize the POC performance using PHY model introduced in the last section.

In CSMA/CA, a radio hears other traffic on overlapped channel and waits for the channel to clean. However in POC, one sender may transmit its packets even through non-clean (partially overlapped) channels. Therefore, we instead propose a time-division (TDMA) scheme, where the transmission time is slotted and links are selected for transmission in different time slots based on their positions and occupied channels. For example, if two links are far from each other, or they use barely overlapped channels, they can be scheduled in the same slot because the interference is small. So given the physical positions of nodes in the mesh network, there are two major decision variables in our MAC algorithm: channel allocation and link scheduling.

Consider a WMN with M nodes and L links. The notations are as below:

d_{ij}	:	distance between node i and j
$l(i,j)$	:	link l from sender i to receiver j
T_{max}	:	the max number of slots in one cycle
$\vec{C} = [c_1, c_2, \ldots, c_M]$	:	channel allocation vector, where c_i is the channe ID used by node i
$\mathbb{Y} = \{y_{lt}\}_{L*T_{max}}$	:	link scheduling matrix, where $y_{lt} = \begin{cases} 1 & \text{if link } l \text{ is active in slot } t \\ 0 & \text{otherwise} \end{cases}$

where $\vec{C}$ and $\mathbb{Y}$ the decision variables.

The link scheduling scheme works periodically. However, there is an upper bound T_{max} for the length of one cycle, which is a parameter to embody the tradeoff between packet delay and system capacity. But T_{max} is only the upper bound for one cycle. After link scheduling, N slots are occupied:

$$N = \sum_{t=1}^{T_{max}} z_t, \qquad z_t = \begin{cases} 1 & \text{if } \sum_{l=1}^{L_1} y_{lt} \geq 1 \\ 0 & \text{otherwise} \end{cases} \ . \tag{6}$$

Then the length of one cycle is reduced to N slots instead of T_{max}.

In a multi-hop mesh network, the traffic characteristics are quite different from single-hop Wireless LAN. Higher sum link capacity do not always lead to better system throughput. For example in Fig. 2, even though link 1, 2, 3 and 5 obtain high link capacities, the traffic is still blocked due to the congestions on link 4, 6 and 7 since node V is the gateway. In other words, we should consider different requirements to get efficient high end-to-end flow throughput in mesh network. The ideal solution is to allocate resource (bandwidth and transmission time) to different links exactly proportional to their requirements. In this case, no valid resource is wasted in multi-hop WMNs due to some bottleneck links. Therefore we introduce one more parameter:

$\vec{R} = [r_1, r_2, \ldots, r_L]$	:	link requirement vector, where r_l is capacity requirement by link l

Our objective is to design the MAC algorithm, where $\vec{R}$ is determined based on the routing protocol in the network layer.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

The problem is formally stated as:
Objective:

$$\max_{\vec{C},\mathbb{Y}} \min_{l=1,2...L} \frac{1}{N}\frac{\sum_{t=1}^{T_{max}} B_{lt}\, y_{lt}}{r_l} \quad (7)$$

such that:

$$\forall\, i,t \quad \sum_{l(i,j)} y_{l(i,j),t} + \sum_{l(k,i)} y_{l(k,i),t} \le 1 \quad (8)$$

where B_{lt} is the capacity of link l at slot t, which can be calculated from (5).
In this paper, we only consider the single-radio non-switched channel problem, which adds the constraint (8). Nonetheless, based on our formulation, it is easy to extend to multiple-radio scenarios, where c_i becomes a vector $\vec{c_i}$ that contains all channels node i can utilize. It is also easy to extend to switched channel scenarios, where c_i becomes c_{it} that may change from time slot to time slot.

III. Heuristic Algorithms for Channel Allocation and Link Scheduling

Given the objective and constraint functions, we need to find out the proper channel allocation vector $\vec{C}$ and link scheduling $\mathbb{Y}$ for a given topology. Unfortunately, both problems are NP-hard [14]. Hence, we approach this problem by decoupling the original problem into two phases and solving it using heuristic algorithms. We first determine channel allocations and then find the optimal link scheduling given the channel allocations determined in the first step.

A. Channel Allocation

To determine channel allocation $\vec{C}$, we adopt the popular Genetic Algorithm (GA) [5], which provides a good framework for finding solutions in a large search space. The algorithm procedure is shown in Algorithm 1.

Initially we randomly produce parent seeds $\vec{C}_{pi}$ with the amount of family size FN (line 5). Then we calculate the optimal system capacity corresponding to each $\vec{C}_{pi}$ through the function *link_scheduling* (line 6), which will be introduced in Sec. III-B. The minimum capacity of FN channel allocation vectors is derived as a bound (line 8) for line 19 .

Next, we perform *CrossOver* and *Mutation* on these parent seeds to derive children, and the children act as parents for the next loop. In the *CrossOver* step (line 12), we exchange halves of two parents $\vec{C}_{pi}$ and $\vec{C}_{p(i+1)}$ to construct one new child $\vec{C}_{ci}$. In the *Mutation* step (line 14-18), we randomly choose two items in $\vec{C}_{ci}$, which mean the channels used by two nodes m and n. If there is a link between node m and n, we decrease the channel distance between m and n; and vice verse. If the capacity according to this child is less than the minimum capacity of parents, we will do the mutation again to update $\vec{C}_{ci}$ (line 19). In the *Selection* step (line 22-30), we choose the best FN channel vectors out of $2 * FN$ vectors ($\vec{C}_{pi}$ and $\vec{C}_{ci}$). When the termination condition is satisfied, we return the best system capacity and corresponding channel allocation vector.(line 35)

Algorithm 1 Channel Allocation Algorithm

1 *input:* Distance matrix $D = \{d_{ij}\}$
2 *output:* channel allocation $\vec{C}$, system capacity f
3 *initial:*
4 for $i = 1 : FN$ {
5 $\vec{C}_{pi}$ =random$[c_1, c_2, \dots, c_M]^i$
6 f_{pi} = link_scheduling $(\vec{C}_{pi}, D)$;
7 }
8 $f_p^{min} = min(f_{pi})$;
9 *loop:*
10 while (! termination condition) do {
11 for $i = 1 : FN$ {
12 $\vec{C}_{ci} = \vec{C}_{pi}[1 \dots M/2] \cup \vec{C}_{p(i+1)}[M+1/2 \dots M]$;
13
14 if $\exists l(m,n)$
15 $|\vec{C}_{ci}^m - \vec{C}_{ci}^n| = min(0, |\vec{C}_{ci}^m - \vec{C}_{ci}^n| - 1)$;
16 else
17 $|\vec{C}_{ci}^m - \vec{C}_{ci}^n| = |\vec{C}_{ci}^m - \vec{C}_{ci}^n| + 1$;
18 f_{ci} = link_scheduling $(\vec{C}_{ci}, D)$;
19 if ($f_{ci} < f_p^{min}$) goto line 14;
20 }
21
22 Median = median$(f_{p1}, \dots, f_{pM}, c_{p1}, \dots, c_{pM})$;
23 $i \leftarrow 1; j \leftarrow 1$;
24 while ($j \le FN$ && $i \le FN$) {
25 if $(f_{pj} \ge Median)\{\vec{C}_{pi} = \vec{C}_{pj}$;
26 f_{pi} = link_scheduling $(\vec{C}_{pj}, D); i \leftarrow i + 1;\}$
27 if $(f_{cj} \ge Median)\{\vec{C}_{pi} = \vec{C}_{cj}$;
28 f_{pi} = link_scheduling $(\vec{C}_{cj}, D); i \leftarrow i + 1;\}$
29 $j \leftarrow j + 1$;
30 }
31 $f_p^{min} = min(f_{pi})$;
32 }
33
34 index = $argmax_i(f_{pi})$;
35 return $\vec{C} = \vec{C}p_{index}; f = fp_{index}$;

The termination condition we chose here is the number of loops (*Loop*). Suppose there are PS possible solutions, and $PS = Ch^M$, where Ch is the number of overlapping channels ($Ch = 11$ in 802.11-base network). We start with FN randomly chosen seeds, and the expectation of the lowest system capacity of these FN seeds is approximately the same as the worst system capacity of all PS solutions. We derive FN children whose system efficiencies are higher than the lowest value of parents. The best FN solutions out of these $2 * FN$ seeds are found at the end of this loop, so the expectation of the lowest system capacity of the new FN solutions is greater than $PS/2$ possible solutions. After $Loop$ steps, the last FN solutions are among the best $PS/2^{Loop}$ solutions. Therefore we can determine the value of $Loop$ specifying how close we need our result to approach the optimal solution.

This result is derived based on some approximations. We assume generations (parents/children) are independent. However in GA, the children inherit their properties from their parents, so they cannot be totally independent. We justify this by noting that children which perform worse will increase the inner loops in line 19. So in the beginning, we can choose $FN * ratio$ seeds ($ratio > 1$), and find the best FN of the $FN * ratio$ seeds as the parents for the first loop. Since children performing worse increases the number of mutations, this will decrease the dependence between the parents and children. In Fig. 6 (Sec. IV-B), the simulation result will show the gap between our result and the optimal solution.

B. Link Scheduling

In this section, we explain the *link_scheduling* function (LS) to accomplish our algorithm. Even with the channel allocation, it is still too difficult to schedule all links at once; instead, we try to schedule links one by one into time slots to achieve high system throughput.

There are L steps in LS. In each step, one link is scheduled into some time slots. Therefore note that L_1 links must have been scheduled after step L_1; meanwhile N_1 time slots are occupied, which is similar to (6). Therefore after step L_1+1, the value of objective function becomes

$$LS_{L_1} = \min_{l=1,2\ldots L_1} \frac{1}{N_1} \frac{\sum_{t=1}^{N_1} B_{lt}\, y_{lt}}{r_l}. \tag{9}$$

In step L_1+1, we will schedule link L_1+1 into one or more slots. There are two possible choices for link L_1+1. (*Choice A*) it may be scheduled into one of the occupied time slots ($1 \rightarrow N_1$); (*Choice B*) it can be scheduled into an idle slot (N_1+1) as long as $N_1+1 \leq T_{max}$.

(*Choice A*) : Let link L_1+1 be scheduled into slot k ($1 \leq k \leq N_1$) with constraint (8), and its capacity be $B^o_{(L_1+1)(k)}$. Let $\vec{S}_k$ be the set of links originally scheduled in slot k, then capacities of links in $\vec{S}_k$ may decrease due to the interference from link L_1+1. Suppose the capacity of link $l \in \vec{S}_k$ decreases from B_{lk} to B^o_{lk}, then the objective function value of link L_1+1 in slot k becomes:

$$\begin{aligned} LS^{(A,k)}_{L_1+1} \;=\; \min\Big(& LS_{L_1},\;\; \tfrac{B^o_{(L_1+1)(k)}}{N_1 * r_{L_1+1}}, \\ & \min_{l \in \vec{S}_k} \tfrac{\sum_{t=1, t\neq k}^{N_1} B_{lt}\, y_{lt} + B^o_{lk}\, y_{lk}}{N_1 * r_l} \Big). \end{aligned} \tag{10}$$

Comparing all possible slots k, the optimal choice for (A) is

$$\begin{aligned} LS^{(A)}_{L_1+1} \;=\; \min\Big(& LS_{L_1},\;\; \tfrac{B^o_{(L_1+1)(j)}}{N_1 * r_{L_1+1}}, \\ & \min_{l \in \vec{S}_j} \tfrac{\sum_{t=1, t\neq j}^{N_1} B_{lt}\, y_{lt} + B^o_{lj}\, y_{lj}}{N_1 * r_l} \Big), \end{aligned} \tag{11}$$

where $j = arg\max_k \{LS^{(A,k)}_{L_1+1}\}$.

(*Choice B*) : Let link L_1+1 be scheduled into an idle slot (N_1+1), and its capacity be $B_{(L_1+1)(k)}$. Then the objective function value becomes:

$$LS^{(B)}_{L_1+1} = \min\left(\frac{N_1}{N_1+1} LS_{L_1}, \quad \frac{B_{(L_1+1)(k)}}{(N_1+1) * r_{L_1+1}} \right). \tag{12}$$

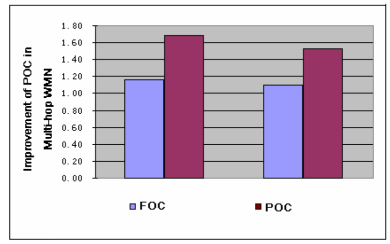


Fig. 3. The improvement of WMN with different link bandwidth requirement

Based on (11) and (12), the objective function value, if link L_1+1 is scheduled in one slot, is

$$LS_{L_1+1} = \max\left(LS^{(A)}_{L_1+1}, \quad LS^{(B)}_{L_1+1} \right). \tag{13}$$

LS_{L_1+1} is the minimal capacity of these L_1+1 links. Therefore in (13), LS_{L_1+1} may be the capacity of link $l (l \neq (L_1+1))$, which means that link L_1+1 obtains enough resource and it is not the bottleneck of the whole network anymore. So link L_1+1 does not have to been scheduled in other slots, and step L_1+1 ends.

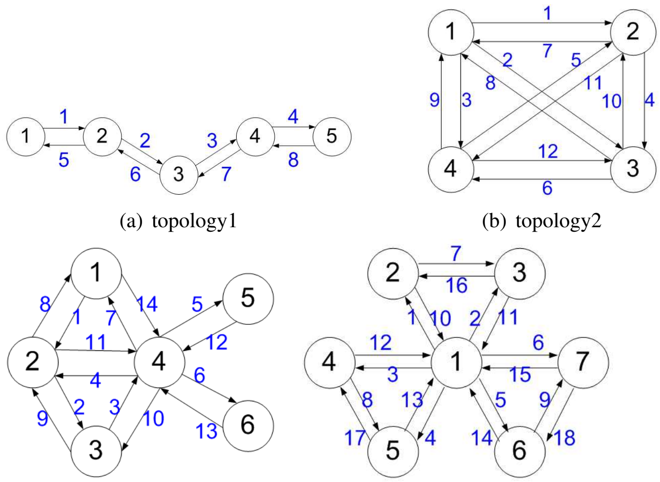
On the other hand, if LS_{L_1+1} is the capacity of link L_1+1, we should schedule more slots to link L_1+1 since it blocks network throughput. Then in addition to slots scheduled for link L_1+1, we loop the procedure above to schedule one more slot to link L_1+1, and derive new LS_{L_1+1}, and so on. So LS_{L_1+1} becomes a function of number of slots given to link L_1+1, n_{L_1+1}. Note that $LS_{L_1+1}(n_{L_1+1})$ is a concave function, and the max value is definitely the final LS_{L_1+1}. We continue this procedure until all L links have been scheduled into proper slots.

IV. SIMULATION

A. End-to-end Flow Requirements

In this section, we evaluate our end-to-end flow algorithm with objective function (7). The simulation results shown in Fig. 3 are based on topology of Fig. 2, where node 5 is the gateway to the Internet. We assume that there are equal traffic flows on the uplink and downlink.

We consider two routing protocols to obtain $\vec{R}$. In the first one (balance-routing), nodes cooperate to distribute traffic evenly through the whole network. In the second one (even-routing), each node will evenly forward its traffic to all output links toward the gateway without considering the traffic balance in the whole network. In Fig. 3, the left two columns come from balance-routing; while the right two columns are based on even-routing. We can observe nearly 30% improvement on system throughput. However this improvement is still not obvious enough since there are only two hops in up- and down-streams. When the number hops increases, POC can provide more benefit. The reason is that in the FOC system, the performance metrics of multi-hop transmissions, such as delay or packet loss, deteriorate quickly. However, POC allows multiple links to transmit at the same time, so the packet delay will be reduced considerably.



(a) topology1 (b) topology2 (c) topology3 (d) topology4

Fig. 4. Four topologies to check the POC performance

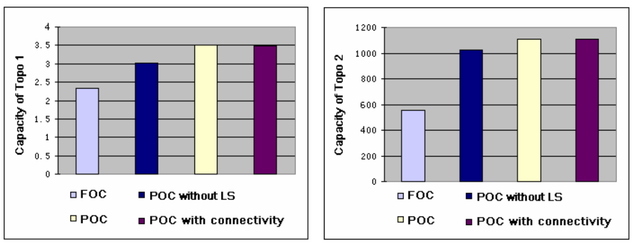


Fig. 5. Performance of POC mechanism in Topologies 1 and 2

B. Effect of Topologies

We evaluate our algorithm by implementing POC on four topologies shown in Fig. 4. Here we change our original objective function to maximize the sum capacity, i.e., $\max_{\vec{C},\mathbb{Y}} \frac{1}{N}\sum_{l=1}^{L}\sum_{t=1}^{T_{max}} B_{lt}\, y_{lt}$, in order to emphasize the effect of topology on system throughput and link capacity.

We display four histograms in the result pictures (Fig. 5 and Fig. 6), in order to compare the impacts of link scheduling and connectivity requirement in mesh networks:

1) FOC : System throughput of FOC mechanism, where two links can transmit simultaneous only if they are out of interference range of each other, and link scheduling is assumed.
2) POC without LS : System throughput of POC without link scheduling
3) POC : System throughput based on our new objective
4) POC with connectivity : System throughput with network connectivity constraint, where $\forall l, \sum_{t=1}^{T_{max}} B_{lt}\, y_{lt} > 0$ (Note: the network connectivity is the primary requirement in mesh network).

Fig. 5 shows that POC improves system capacity significantly. Especially in the topology of Fig. 4(b), the system capacity of POC doubles compared with that of FOC. Comparing column 2 and 3, we find that in POC, link scheduling is very important. Comparison of column 3 and 4 shows that the network connectivity constraint does not degrade system capacity much.

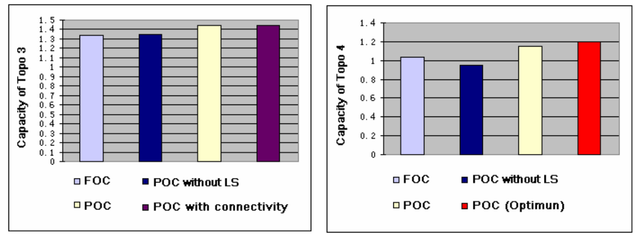


Fig. 6. Performance of POC mechanism in Topologies 3 and 4

However, the capacity improvement of POC in Fig. 6 is limited. Especially in the topology of Fig. 4(c), the capacities of FOC and POC are comparable. This is due to bottleneck nodes (node 4 in Fig. 4(c), and node 1 in Fig. 4(d)).

The main motivation for employing POC is that we can schedule more simultaneous transmissions. However, consider the example of Fig. 4(d), 12 links are associated with node 1. If one of those links is transmitting, the other 11 links do not transmit due to constraint (8). In this case, we cannot reduce the total number of slots as was the case in topologies of Fig. 4(a) and 4(b). Therefore the influence of POC is quite limited.

Note that in the result figure for topology of Fig. 4(d), we replace the fourth histogram with the optimal system capacity, obtained by searching all feasible solutions. Clearly there is a gap between the results from GA algorithm and the optimal result from exhaustive searching. There are 11 overlapped channels in ISM band; in this topology, there are 7 nodes; in the GA algorithm, 10 loops are executed. Therefore our solution is approximately among the best $11^7/2^{10} \approx 10^4$ solutions. So it is not surprising that there is a gap between our solution and the optimal capacity. In order to reduce the gap, we can set the number of loops in GA to $Loop(M)$, which is a linear function of M (number of nodes in the network):

$$Loop(M) = M * \log_2 Ch - \log_2 \rho, \quad (14)$$

where ρ is one integer. Then our solution is approximately among the best ρ solutions.

Based on the results of Fig. 5 and Fig. 6, we see that POC works well for "*symmetric*" topologies where all of the nodes have similar degrees, but it is comparable to FOC for "*asymmetric*" topologies where one or two nodes have much higher degrees than others. This is due to the fact that in the symmetric topologies, more potential links can transmit simultaneously. The performance of POC should not be inferior to that of POC— POC solution sets include FOC sets, or FOC can be considered as one special case of POC. But the algorithm may not find the optimal POC solution, which will be shown in IV-C.

In addition, Fig. 7 shows the capacities of individual links on topology of Fig. 4(a). We find that the improvement gained by POC is not evenly distributed among all links. The whole system improves at the expense of decreasing the capacities of some links. The unfair allocation among links may block multi-hop transmissions in mesh network, so it is

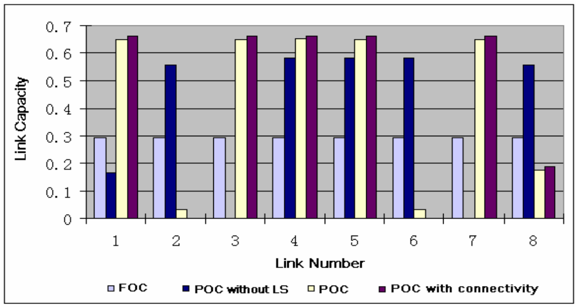


Fig. 7. The capacities of individual links in topo 1

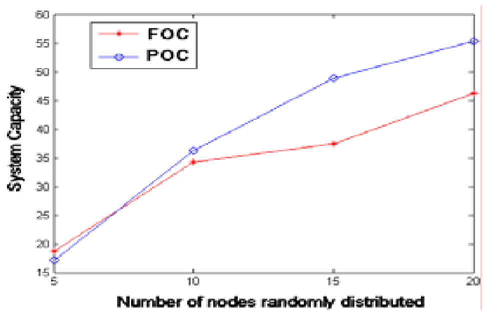


Fig. 8. The System Capacity VS. Node Density.

not efficient only to maximize total link capacities. This is the basic motivation for using end-to-end flow requirements as the metric in our algorithm.

C. Effect of node density and distribution

In addition to the fixed topologies in the section IV-B, we evaluate POC under various randomly generated topologies. We construct a square area, whose perimeter is 24 times of the node transmission range, and place M nodes randomly in this area following a uniform distribution.

The results are shown in Fig. 8 based on the average of multiple simulations. We find that POC is more effective for higher node density. If the nodes are distributed too sparsely, most of the links can transmit simultaneously even if all of them use the same channel because they are likely to be out of interference range of each other. Under these conditions, the POC mechanism cannot provide much improvement over FOC. However, when the nodes density is high, resulting in more link contentions, POC mechanism can find much better re-use of space and spectrum. Again, if the nodes are distributed more evenly (leading to more uniform node degrees), POC performs better as previously observed.

In addition, when the node density is very low, FOC seems to perform better than POC because the GA performs fewer iterations and has less chance of returning a good result.

V. Conclusion

For 802.11 wireless networks, people tend to choose orthogonal channels to reduce interference, therefore FOC mechanism is used to maintain connectivity in WMNs. In this paper, we investigate its complement, POC. POC has the potential of increasing capacity in WMNs by allowing more links to transmit simultaneously.

The challenge in using POC is the combination of channel allocation and link scheduling. In general, making a tradeoff between the simultaneous transmission and interference is an NP-hard optimization problem. So we adopt heuristic algorithms to search for a sub-optimal solution. We divide the original problem into two components and design algorithms to solve them independently..

Since the traffic characteristics in multi-hop WMNs are quite different to one-hop 802.11 network, we design our algorithm to meet end-to-end flow transmission constraints by considering different link requirements in the network. This improves the system throughput to and from the Internet, instead of the sum of link capacities in the WMNs.

In addition to the proposed algorithms, we also discuss some other factors that influence the performance of POC, such as topology, node density, and node distribution. We find that POC works better in multi-hop wireless networks that are symmetric, with a high density of evenly-distributed nodes.

References

[1] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou. A multi-radio unification protocol for ieee 802.11 wireless networks. In *BROADNETS*, 2004.

[2] I. F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Computer Networks*, 47(4), 2005.

[3] M. Alicherry, R. Bhatia, and L. E. Li. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *MobiCom*, 2005.

[4] P. Bahl, R. Chandra, and J. Dunagan. Ssch: Slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks. In *MobiCom*, 2004.

[5] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.*, 35(3), 2003.

[6] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom*, 2004.

[7] P. Gupta and P. Kumar. Capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2), 2000.

[8] P. Kyasanur and N. H. Vaidya. Routing and interface assignment in multi-channel multi-interface wireless networks. In *IEEE WCNC*, 2005.

[9] A. Mishra, E. Rozner, S. Banerjee, and W. Arbaugh. Exploiting partially overlapping channels in wireless networks: Turning a peril into an advantage. In *ACM/USENIX Internet Measurement Conference*, 2005.

[10] A. Mishra, E. Rozner, S. Banerjee, and W. Arbaugh. Using partially overlapped channels in wireless meshes. In *Wimesh*, 2005.

[11] A. Mishra, V. Shrivastava, S. Banerjee, and W. Arbaugh. Partially overlapped channels not considered harmful. *SIGMETRICS Perform. Eval. Rev.*, 34(1), 2006.

[12] A. Rad and V.Wong. Joint optimal channel assignment and congestion control for multi-channel wireless mesh networks. In *ICC*, 2006.

[13] B. Radunovic and J. L. Boudec. Rate performance objectives of multi-hop wireless networks, 2004.

[14] A. Raniwala, K. Gopalan, and T. cker Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 8(2), 2004.

[15] N. Shacham and P. J. King. Architectures and performance of multi-channel multi-hop packet radio networks. *IEEE Journal on Selected Area in Communication*, 5(6), 1987.

[16] J. So and N. H. Vaidya. Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. In *MobiHoc*, 2004.