2008

A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks

Kwan-Wu Chin
University of Wollongong, kwanwu@uow.edu.au

Recommended Citation

Chin, Kwan-Wu: A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks 2008.
https://ro.uow.edu.au/infopapers/3124

A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks

Abstract

Wireless routers equipped with smart antennas are capable of forming beams to neighboring devices to transmit/receive multiple packets simultaneously, hence achieving high network capacity. This however is dependent on the link scheduling algorithm employed by these routers. To this end, we describe a simple link activation algorithm that tradeoffs path length to increase network capacity. We show via analysis and simulation that the proposed algorithm improves network capacity and lowers end-to-end delay despite a slight increase in path length.

Disciplines

Physical Sciences and Mathematics

Publication Details

K. Chin, "A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks," in IEEE International Conference on Communications, 2008, pp. 1-5.

A New Link Scheduling Algorithm for Concurrent Tx/Rx Wireless Mesh Networks

Kwan-Wu Chin
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong
Northfields Avenue, NSW, Australia
kwanwu@uow.edu.au

***Abstract*— Wireless routers equipped with smart antennas are capable of forming beams to neighboring devices to transmit/receive multiple packets simultaneously, hence achieving high network capacity. This however is dependent on the link scheduling algorithm employed by these routers. To this end, we describe a simple link activation algorithm that tradeoffs path length to increase network capacity. We show via analysis and simulation that the proposed algorithm improves network capacity and lowers end-to-end delay despite a slight increase in path length.**

I. INTRODUCTION

Wireless mesh networks offer significant benefits in environments that lack communication infrastructures. Example applications include ad-hoc video surveillance systems, and facilitating communications in public safety emergency scenarios, all of which require high capacity links in order to transport real-time audio/video streams.

A promising but challenging approach to increase the capacity of mesh networks is to equip routers with smart antennas. In such networks, routers have the ability to focus their transmission energy or electromagnetic beam at specific location/direction. Moreover, the nulling and beam steering capabilities of such antennas mean a router has a longer range and causes little RF pollution. Thereby, improving gains and lowering bit error rates during packet reception.

A key property of the aforementioned routers is their ability to transmit or receive n distinct packets from n neighbors simultaneously, viz. spatial division multiple access (SDMA). This is a marked improvement over existing omni-directional based wireless mesh networks, where routers can only transmit/receive to/from one neighbor at a given time. However, this improvement can only be achieved in conjunction with a scheduling algorithm. Specifically, given a network topology, the scheduler must maximize the number of concurrent transmissions/receptions at a given point in time.

To this end, we present a new link scheduling algorithm that tradeoffs path length to increase network capacity. Specifically, our algorithm eliminates cliques of size three to reduce the number of colors or frame size of a given wireless mesh network. Our results show that the proposed algorithm dramatically increases network capacity and reduces end-to-end delay.

The remainder of the paper has the following structure. Section II and III present the system under consideration, and the link activation problem respectively. Then, we present our algorithm in Section IV, and proof some of its properties in Section V. We then present the results from our simulation study in Section VI. Section VII review related works, and Section VIII concludes the paper.

II. SYSTEM ARCHITECTURE

Smart antennas [5][3] technologies are becoming popular. Devices using smart antennas rely on beamforming algorithms to provide maximum gains toward a device and form nulls toward interferers. An example system is shown in Figure 1, where router N_a has beamformed towards N_b and N_c. Hence, it is able to transmit or receive from both N_b and N_c concurrently. These two beams are achieved by deriving a steering vector using constrained optimization techniques, where the channel coefficients of each neighboring device are used to derive the weights of each antenna element. Note, the steering vector derivation is typically facilitated by the unique training sequence of each device. In theory, a k-element array is able to provide k spatial channels or null out $k-1$ interferers [3]. The former allows one to many connections whereas the later enables neighboring devices to transmit simultaneously; as opposed to being blocked when devices use omni-directional antenna.

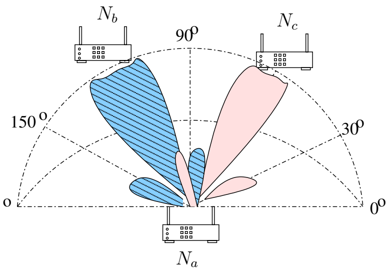


Fig. 1. Beamforming using smart antennas.

Besides smart antennas, another method to achieve concurrent transmit or receive is to employ multiple network interfaces, each connected to a directional antenna. In [8], Raman et al. equip access points with multiple IEEE 802.11b interfaces and connect each of them to a parabolic grid antenna. Raman et al. then adjust these access points' transmission power in order to transmit/receive packets from multiple neighbors simultaneously. Note, in their system, an access point can only transmit to its neighbors concurrently provided that these neighbors are at a given distance away and the angle between them are beyond a certain threshold; as dictated by antenna side-lobe characteristics and beamwidth. An attractive feature of their system is that each interface can be tuned to a different channel, which they exploit in a subsequent work [7] to meet variable traffic demands.

To take advantage of the concurrent transmission or reception capability, each router uses spatial reuse TDMA [6] to maximize the number of concurrent transmissions in a given time slot. Figure 2 shows an example TDMA-based wireless mesh network. First consider the case where each router is equipped with an omni-directional antenna. The sets of links that can transmit concurrently at a given time slot include $\{L_1, L_{10}\}$, $\{L_4, L_9\}$ and $\{L_4, L_7\}$. On the other hand, with smart antennas, routers N_c, N_e and N_f can transmit on all their respective antennas concurrently, i.e., links $\{L_1, L_3, L_6\ L_7, L_9\}$ can be activated in the same time slot. In the following time slot, transmitting devices prepare to receive, thereby activating the links $\{L_2, L_4, L_5, L_8, L_1\}$. From these examples, we see that the omni-directional and smart antennas case affords two and five links respectively. In other words, the use of smart antennas has increased the capacity of the network shown in Figure 2 by 250%.

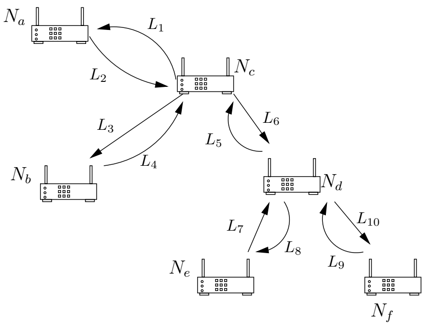
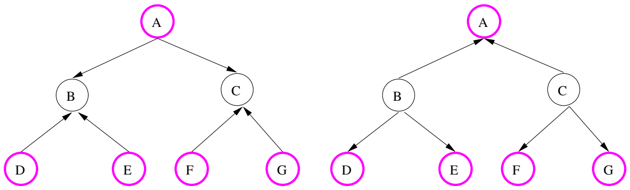


Fig. 2. A Wireless Mesh Network

It is important to note that smart antennas enable routers to selectively disable their links to a subset of neighbors. Conversely, routers can form nulls toward interfering or transmitting neighbors. This is in contrast to routers using an omni-directional antenna where their transmissions interfere with all neighboring nodes. As we will see later in Section IV, our algorithm exploits this smart antenna property to disable links in a clique, and hence differentiates itself from existing omni-directional based link/node scheduling algorithms.

III. THE PROBLEM

In the proposed system, the fundamental goal of a scheduler is to maximize the number of links scheduled within a time slot. Consider Figure 3. At time t_1, we see that nodes B and C are receiving packets concurrently from parent A and their respective child nodes. At t_2, nodes B and C start transmitting whilst other nodes receive. From a graph coloring perspective, since a tree is a bipartite graph, only two colors or slots are required to ensure each link gets a transmission and reception opportunity. Therefore, bipartite network topologies yield the shortest schedule.



(a) Slot t_1 (b) Slot t_2

Fig. 3. Scheduling links in a tree topology

In practice, the k_5 topology shown in Figure 4 may represent reality better. Unfortunately, it requires eight slots to ensure all links have a chance to transmit and receive. For example, in slots t_1 and t_2, node-A can transmit/receive to/from nodes B to E. After t_2, all links incident on node-A have been serviced. At slot t_3, node C, D or E is then selected to transmit. Working through the example, a total of eight slots will be required to ensure all links have at least one transmit and receive opportunities. Note that a link can be allocated additional slots in a given frame if it does not interfere with other scheduled links in a given time slot.

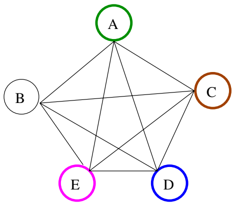


Fig. 4. K_5

In the example above, we assumed each node has less than k neighbors. Recall that a node only has k-elements in its smart antenna. This means if a node has more than k neighbors, more slots may be needed to ensure all links/neighbors receive transmission and reception opportunities. Interestingly, it may be worthwhile to only consider a subset of k links in each time slot. Consider Figure 5. If we schedule according to the number of links a node has, say $k = 3$, six time slots are required. On the other hand, if we choose to activate only a subset of a node's links, say two out of three links, the box topology requires only four slots. Notice that the resulting subgraphs are bipartite.

In summary, this paper addresses the following problem. Given a wireless mesh network running spatial TDMA, determine a maximal concurrent transmit and receive schedule that satisfies the following constraints:

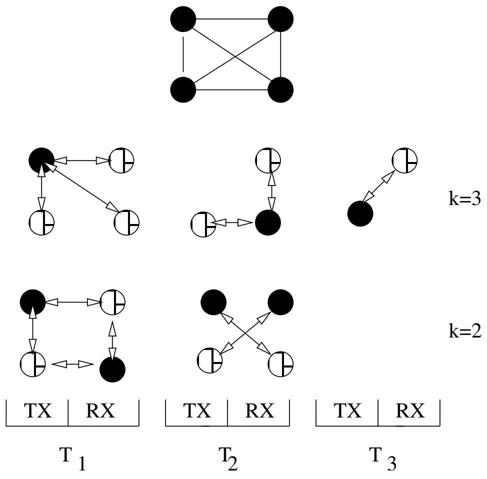


Fig. 5. Box Topology

- Each link must receive two time slots; transmit and receive.
- A node can either transmit or receive on all links. In other words, a node is not allowed to receive on some links and use its remaining links to transmit.
- Each node is only allowed to form a maximum of k concurrent links, where k corresponds to the number of antenna elements.

Besides the constraints above, a scheduler may consider the traffic demand of each link and allocates additional transmission/reception opportunities to those with high traffic load. As an example, consider the $k = 3$ scenario in Figure 5. At time t_2, it is also possible to include the link to the top left node even though that link has received service at time t_1. Similarly, at time t_3, previously scheduled links can be given additional time slots according to their traffic load. We leave this as a future work.

IV. PROPOSED ALGORITHM

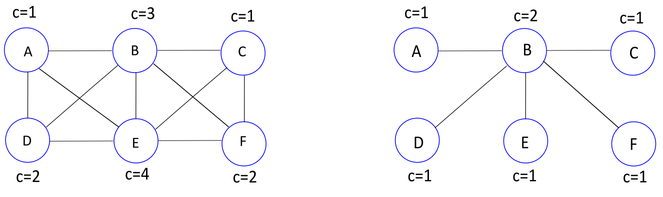
Given a connected graph $G = (V, E)$, where V and E denote the set of vertices and edges respectively, we now propose an algorithm that reduces the number of colors required for a given graph at the expense of a moderate increase in path length. Note, our algorithm is independent of any graph coloring algorithm. Moreover, all nodes are stationary.

Algorithm 1 shows the steps required to increase the capacity of a wireless mesh network. The key observation is that removing cliques of size three reduces the chromatic number of a graph. The algorithm starts at the vertex/node with the highest degree, V^h. If there are nodes with a similar degree, the algorithm selects one at random. It then visits the children of V^h and determine whether there is an edge connecting any of its children. If there is, the algorithm removes the edge connecting the children in question. After that, the algorithm selects the next node with the highest degree. Bear in mind that the graph now contains a smaller number of edges. After visiting all nodes, the algorithm calls a graph coloring algorithm. Finally, each node is allocated a time slot corresponding to its assigned color, meaning the frame length is equal to the chromatic number of G.

```
input : G(V, E)
output: Vertices Color
$\nu = V$
while $\nu \neq \{\}$ do
    $V^h$ = MAX_DEG($\nu$)
    $V_c$ = CHILDREN($V^h$)
    for $c \in V_c$ do
        for $k \in V_c$ do
            if EDGE(c, k) $\in E$ then
                $E \setminus$ EDGE($c$, $k$)
            end
        end
    end
    $\nu \setminus V^h$
end
return (COLOR(G))
```

Algorithm 1: Proposed algorithm. The function MAX_DEG and CHILDREN returns the vertex with the highest degree, and the set of vertices adjacent to V^h respectively.

Figure 6 shows what happens when the proposed algorithm is applied to topology 6(a), which requires four colors. Our algorithm starts at Node-B. After iterating through Node-B's children, the algorithm determines that there is an edge connecting the following children: C-F, E-F, A-D, A-E, E-C, and D-E. As a result, these edges are removed from the topology, yielding the topology 6(b). The resulting topology only requires two colors, a marked improvement over the original topology. Notice that the nodes connected by the edges removed by our algorithm can only reach each other via Node-B. In other words, our algorithm has tradeoff path length for network capacity.



(a) Before (4 colors)

(b) After (2 colors)

Fig. 6. Example of the proposed algorithm in operation. The value of c above each node corresponds to a color number.

V. ANALYSIS

We now present various properties of our algorithm, which we refer to in the following discussion as *Algo-1*.

Lemma 5.1: Algo-1 removes all cliques in graph G.

Proof: By definition, in a clique of size N, a vertex has connectivity to $N - 1$ vertices. When *Algo-1* encounters a vertex that is part of a clique, say ν, it removes all edges connecting ν's children. In other words, other than the edge connecting ν to its children, all other edges are removed, specifically $\frac{N(N-1)}{2} - (N-1)$ edges. This means *Algo-1* collapses the clique into a tree; i.e., from N to two colors. Therefore, as *Algo-1* visits all vertices in G, and that G is connected, *Algo-1* removes all cliques in G. ■

Corollary 5.2: It is easy to see that the hop count of each vertex that was in the clique before increases by one, since the vertices has to route their packets via ν.

Corollary 5.3: The removal of edges does not result in a cycle, since the resulting topology is a tree. This is important for the reason described next.

A factor that increases the chromatic number of G are odd cycles. An odd cycle of N vertices require three colors, as opposed to two for even cycles. Figure 7 shows a topology with an odd cycle. One approach to reduce the number of colors is by disabling a link in the cycle. For example, if we remove the link B-F, then the topology is bipartite, hence requiring only two colors. Unfortunately, this approach is counter-productive. For example, instead of a two hops path from node A to F, the resultant path length is five. In the worst case, an existing path length increases from γ to $(\gamma + N - 2)$, which erodes the performance gains obtained from reducing the chromatic number of a given graph/topology. Therefore, we omit this approach from consideration.

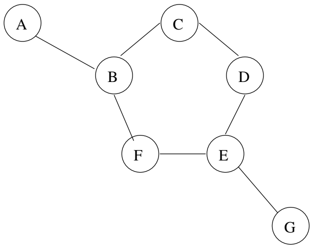


Fig. 7. An example topology with an odd cycle.

Lemma 5.4: The maximum end-to-end delay of a path with length γ over a graph G with chromatic number c is $(c-1)(\gamma-1) + c$ slots.

Proof: We will use Figure 6(a), since transmitting packets over topologies with cliques yield the worst delay. This is because of the following reason. Consider a packet making its way from node-A to node-C. Assume that a packet transmission takes one time slot and there is no queueing delay. In the worst case scenario, node-A needs to wait for three or $c - 1$ time slots before it is allowed to transmit. Therefore, the packet takes four time slots to egress node-A. However, at node-B, the packet only needs to wait for $c-2$ time slots before receiving permission to transmit. This is because one time slot has been consumed by node-A, and another belongs to node-B. In total, the packet incurred a delay of seven time slots before arriving at node-C. In general, we have $(c-1)(\gamma-1) + c$ slots, since the first hop incurs a delay of c, and subsequent hops, i.e., $\gamma - 1$, has a delay of $c - 1$. Note, in the best case scenario, if node-A and B receive their time slot upon packet arrival, it only takes two time slots to reach node-C. ■

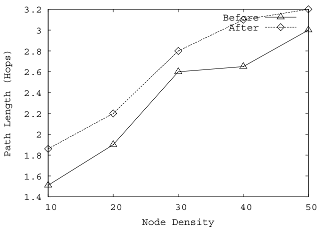


Fig. 8. Path Length.

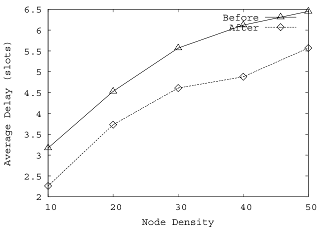


Fig. 9. Delay.

VI. SIMULATION

To study the performance of our algorithm in various topologies, we use MatGraph [9] – a Matlab toolkit for working with simple graphs. Although the toolkit comes with five graph coloring algorithms, we only use the "optimal" algorithm given its ability to yield the minimal number of colors for a given graph. In practice, given that nodes in wireless mesh networks are generally static, we expect our algorithm to be run infrequently.

Our experiments involve varying node density and degree. For each experiment, we record the resulting chromatic number of a given topology before and after applying the proposed algorithm. We also compute the average end-to-end packet delay for each topology. In other words, once a schedule is computed, we inject packets from randomly chosen source and destination nodes, and record the average end-to-end delay incurred by these packets. Here, the speed in which these packets traverse the network is determined by the color of nodes on the shortest path from their respective source to destination node.

A. Node Density

We first investigate the impact of node density. Figures 8 and 9 show the average end-to-end delay and path length over topologies with 10 to 50 nodes. In the former figure, we observe a slight increase in path lengths. Despite that, nodes experience better end-to-end delay since the resulting schedule or chromatic number after our algorithm is applied is shorter/smaller.

B. Node Degree

Next, we study the impact of node degree. The results are shown in Figures 10 and 11. Initially, we see the benefits of increasing node degree since doing so enables shorter paths to be constructed, thereby reducing end-to-end delay. However, once nodes have four or more neighbors, they start to experience increasingly higher delays. On the other hand, when our algorithm is applied, nodes see a gradual decrease in delay. Again, this is due to our algorithm's ability to reduce the chromatic number of a network/graph.

VII. RELATED WORK

Many researchers have investigated link/node activation algorithms for wireless ad-hoc networks. They have however only considered nodes with omni-directional antenna. Interested readers are referred to [1] and references therein for

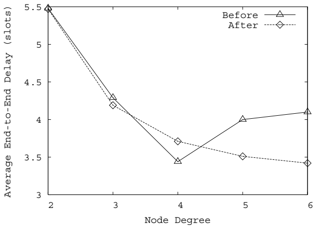


Fig. 10. Delay.

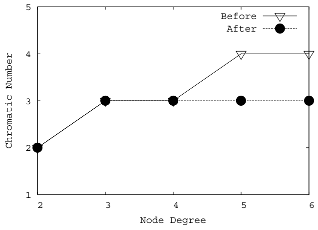


Fig. 11. Color.

more information. The key drawback of these works is that they are interference limited. In other words, they have poor spatial reuse since a receiving node blocks all its neighbors from transmitting. Conversely, a transmitting node prevents its neighbors from receiving. Thereby, severely limiting the number of concurrent transmissions or receptions that can occur within a wireless mesh network. Lastly, these works cannot be applied directly since they assume a node can only transmit/receive one packet at a given time, whereas the nodes in our system can transmit/receive up to k packets at a time.

To date, only a handful of works have considered the concurrent transmit/receive abilities of smart antennas. Raman et al. [8] presented a wireless mesh network that interconnects rural villages in India. Each access point runs a MAC called 2P, which requires access points to be either in the transmit or receive mode. Raman et al. then proposed an algorithm to construct a tree topology that meets various criteria, namely transmission range, angle between neighbors, and hop count or tree depth. A key limitation of Raman et al. [8]'s work is the need for a bipartite graph. In a subsequent work, Raman [7] sought an algorithm to color the edges of a network topology with three colors[1] and construct bipartite subgraphs consisting of links with similar traffic requirements. Unfortunately, such algorithm was found to be NP-complete. Henceforth, Raman proposed and studied various heuristics for forming subgraphs that minimize the discrepancies between desired and allocated link capacity. Sundaresan et al. [10] exploit the nulling capability of smart antennas to maximize the number of concurrent transmissions at a given point in time. The authors showed that scheduling transmissions according to the number of cliques a node belongs to in a contention graph ensures nodes have the maximal degrees of freedom to transmit/receive packets, which otherwise would have been used to form nulls toward interfering neighbors. Bao et al. [2] present a distributed algorithm that makes use of two hops neighbor information to activate links according to their priority. When a link is activated for transmission, the algorithm determines whether there are other links in its two hops range that can be activated simultaneously. The aforementioned works have not considered trading off path length for capacity. In this respect, we believe our algorithm offers these existing works an optimization to improve network capacity further.

There have also been works, e.g., [4], that consider nodes with only a single beam; steerable or switched beam antennas. These works aim to maximize the number of node pairs or non interfering links at a given point in time with the constraint that a node can only transmit or receive from a given neighbor, a marked difference to the system considered in this paper.

VIII. CONCLUSION

This paper has presented a simple algorithm to reduce the frame size of a spatial TDMA based concurrent transmit/receive wireless mesh network. In particular, the algorithm exploits the observation that path length can be sacrificed to some extent in order to increase network capacity.

There are many avenues for future works. One of which is to consider traffic demands, and schedule links with a higher load more frequently. Apart from that, we are investigating the possibility of designing a distributed MAC based on our algorithm. Lastly, we will consider the case where nodes degree exceeds the number of antenna elements.

IX. ACKNOWLEDGMENT

The author like to thank Nick Kosiak for implementing the proposed algorithm in MatGraph and compiling the results.

REFERENCES

[1] L. Bao and J. Garcia-Luna-Acceves. *Algorithms and Protocols for Wireless and Mobile Networks*, chapter Distributed Channel Access Scheduling for Ad Hoc Networks. Chapman and Hall, CRC, 2004.

[2] L. Bao and J. Garcia-Luna-Aceves. Transmission scheduling in ad hoc networks with directional antennas. In *Proceedings of the 7th ACM/IEEE Internation Conference on Mobile Computing and Networking (MOBICOM'2001)*, Atlanta, USA, Sept. 2002.

[3] M. Cooper and M. Goldburg. Intelligent antennas: Spatial division multiple access. Annual Review of Communication, 1996.

[4] J. Groönkvist. Assigning strategies for spatial reuse TDMA. Master Thesis, Mar. 2002. Sweden Royal Institute of Technology, TRITA-S3-RST-0202, ISSN 1400-9137.

[5] J. C. L. Jr and T. S. Rappaport. *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*. Prentice-Hall, 1999.

[6] R. Nelson and L. Kleinrock. Spatial TDMA: A collision-free multihop channel access protocol. *IEEE Transactions on Communications*, COM-33(9):934–945, Sept. 1985.

[7] B. Raman. Channel allocation in 802.11 based mesh networks. In *IEEE INFOCOM*, Miami, FL, USA, Mar. 2005.

[8] B. Raman and K. Chebrolu. Design and evaluation of a new MAC for long-distance 802.11 mesh networks. In *The 11th Intl. Conference on Mobile Computing and Networking (MOBICOM)*, Colongne, Germany, Aug. 2005.

[9] E. R. Scheinerman. Matgraph 1.7. http://www.ams.jhu.edu/ ers/matgraph.

[10] K. Sundaresan, R. Sivakumar, M. A. Ingram, and T.-Y. Chang. A fair MAC protocol for ad-hoc networks with MIMO links. In *IEEE INFOCOM*, Hong Kong, China, 2004.

---

[1] The IEEE 802.11b has three non-overlapping channels