Hybrid Multi-Channel Multi-Radio Wireless Mesh Networks

Yong Ding, Kanthakumar Pongaliur, Li Xiao
Department of Computer Science and Engineering
Michigan State University
Email: {dingyong, pongaliu, lxiao}@cse.msu.edu

***Abstract*—Many efforts have been devoted to maximizing network throughput in a multi-channel multi-radio wireless mesh network. Current solutions are based on either pure static or pure dynamic channel allocation approaches. In this paper, we propose a hybrid multi-channel multi-radio wireless mesh networking architecture, where each mesh node has both static and dynamic interfaces. We first present an Adaptive Dynamic Channel Allocation protocol (ADCA), which considers optimization for both throughput and delay in the channel assignment. In addition, we also propose an Interference and Congestion Aware Routing protocol (ICAR) in the hybrid network with both static and dynamic links, which balances the channel usage in the network. Compared to previous work, our simulation results show that ADCA reduces the packet delay considerably without degrading the network throughput. Moreover, the hybrid architecture shows much better adaptivity to changing traffic than pure static architecture without dramatic increase in overhead.**

I. INTRODUCTION

Wireless mesh networking [1] has attracted great research interest recently. It has become a promising technology that has the potential to enable many useful applications. One major problem facing wireless mesh networks (WMN) is the capacity reduction due to wireless interference [2] [3] [4]. Technology advances have made it possible to equip a wireless mesh router with multiple radios, which can be configured to different channels, and thus reducing network interference. Therefore, a major challenge in multi-channel wireless mesh networks is the allocation of channels to interfaces of mesh routers so as to maximize the network capacity.

There are currently two approaches of channel allocation, that is, static approach and dynamic approach. In static channel allocation, each interface of every mesh router is assigned a channel permanently. In dynamic channel allocation, an interface is allowed to switch from one channel to another channel frequently. Both strategies have their advantages and disadvantages.

Static strategies do not require interfaces to switch channels, and thus have lower overhead. However, they depend on stable and predictable traffic patterns in the network. For example, [5] [6] [7] require that the exact traffic profile is known beforehand, while [8] [9] assume knowledge of statistical traffic patterns. Dynamic strategies, such as [10] [11] [12] [13], require frequent channel switching, and thus have higher overhead than static strategies. However, as the channel allocation can be changed with the changing traffic, dynamic strategies are more appropriate when the network traffic changes frequently and is unpredictable.

In the real environment, the overall traffic profile is usually complex. It not only contains some predictable traffic, e.g., a large amount of traffic from end-users to the Internet through gateways, but also contains a considerable amount of unpredictable peer-to-peer traffic between end-users due to the emerging new applications within the community. Due to the inflexibility of pure static channel allocation and the high overhead of pure dynamic channel allocation, we propose a hybrid architecture in this paper, which combines the advantages of both approaches. In this architecture, one interface from each router uses the dynamic channel allocation strategy, while the other interfaces use the static channel allocation strategy. The links working on static channels provide high throughput paths from end-users to the gateway while the links working on dynamic channels enhance the network connectivity and the network's adaptivity to the changing traffic. Therefore, this hybrid architecture can achieve better adaptivity compared to the pure static architecture without much increase of overhead compared to the pure dynamic architecture.

In this paper, we discuss several important issues in the hybrid wireless mesh network. (1) The system architecture, where one radio works as dynamic interface and the other radios work as static interfaces in each node. (2) The channel allocation for dynamic interfaces. MMAC [11] is currently one of the most efficient dynamic channel allocation protocols. However, the channel assignment in MMAC is optimized only for network throughput. We propose an Adaptive Dynamic Channel Allocation protocol (ADCA), which considers both throughput and delay in the channel assignment. Compared with MMAC, ADCA is able to reduce the packet delay without degrading the network throughput. (3) Routing decision in the network. In the hybrid structure, we have static links and dynamic links, both of which can be used to transmit data. We propose an Interference and Congestion Aware Routing protocol (ICAR), which aims at balancing the channel usage over the network, thereby improving the network throughput.

The rest of the paper is organized as follows. We introduce the system model in Section II. In Section III and IV, we present our dynamic channel allocation protocol and the routing algorithm in the hybrid wireless mesh network. We evaluate our protocols in Section V, and finally conclude our work in Section VI.

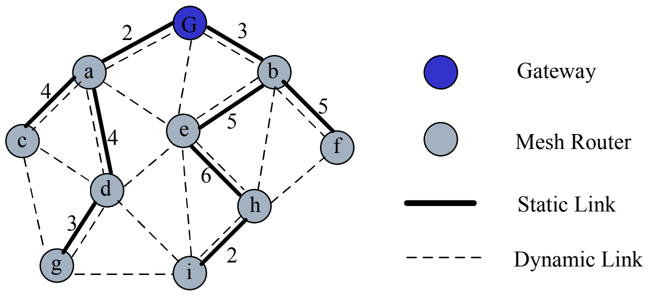


Fig. 1: The Hybrid WMN Architecture

II. SYSTEM MODEL

In this paper, we propose to use the hybrid architecture to achieve both adaptivity to changing traffic and low channel switching overhead. Let $G(V, E)$ be the network topology, where V is the set of mesh routers and E represents pairs of mesh routers that are within radio communication range. Assume each mesh router has multiple interfaces. In the hybrid architecture, we let one interface of each mesh router be able to switch channel frequently, and the other interfaces work on fixed channels. In the rest of this paper, we call the former interface as *dynamic interface*, and the latter as *static interface*.

Fig.1 illustrates a hybrid multi-channel multi-radio wireless mesh network. Most mesh nodes including the gateway have 3 interfaces, and a few boundary nodes $\{c, g, i, f\}$ have 2 interfaces. For each mesh node, one interface works as dynamic interface, and the others work as static interfaces.

The channel allocation of static interfaces aims at maximizing the throughput from end-users to gateways, which usually constitutes a major portion of the traffic in the network. For this purpose, heuristic algorithms, such as [8], could be used. In the algorithm, a load balanced tree is constructed for each gateway. The goal of the tree construction is to allocate bandwidth fairly to each user with regard to the user-gateway throughput. After the topology has been constructed, each link can then be assigned channels. The links closer to the gateways are given higher priority to be allocated with less congested channels. In Fig.1, the tree topology is shown in bold lines, and we call these links as *static links*. The channels assigned to static links are also shown in the figure.

Dynamic interfaces work in an on-demand fashion. Two dynamic interfaces that are within radio transmission range of each other are able to communicate by switching to a same channel when they have data to transmit. We call these links as *dynamic links*. Fig.1 illustrates all possible dynamic links in dotted lines. Note that each dotted line only implies that the pair of nodes are able to communicate because they are within radio transmission range of each other, but they need to switch to the same channel before they can transmit data.

In the following sections, we focus on two problems: 1) How dynamic interfaces schedule channels to transmit data? We propose a dynamic channel allocation protocol, which optimizes for both throughput and packet delay. 2) How to route traffic in the hybrid network? We propose a routing mechanism, which utilizes both static and dynamic links in the selection of routes.

III. ADAPTIVE DYNAMIC CHANNEL ALLOCATION

The throughput of wireless mesh networks can be dramatically increased by utilizing multiple channels instead of a single channel. MMAC [11] is a dynamic channel allocation protocol for wireless mesh networks, where each node has a single dynamic interface. In MMAC, the time is divided into fixed-length intervals, each of which consists of control interval and data interval. In the control interval, any two nodes that have data to transmit communicate on a default channel (or control channel) to negotiate the channel to use in the data interval. In the data interval, pairs of nodes transmit and receive data on the negotiated channels (or data channel).

MMAC improves the network capacity at the cost of increasing packet delivery delay. When the traffic load is below saturation, MMAC may cause unnecessary packet delay, which can be illustrated by the examples in Fig.2. 1) In Fig.2(a), assume A has some data to send to C. Let the maximum bit rate of each interface be R, and assume the traffic rate is less than $R/2$. According to MMAC, in the first time interval t_1, the packets are transmitted from A to B, and then in the second interval t_2, the packets are transmitted from B to C. Therefore, the packet delay is around two intervals. On the other hand, if we assign A, B and C with the same channel, and use 802.11 to resolve contention in this subnetwork, the packets can be transmitted from A to C within one interval. 2) In Fig.2(b), assume A has some data to both B and C, with the aggregate traffic rate of less than R. We can see that MMAC still needs two intervals to complete the transmission, while it can actually be done in one interval by assigning the same channel to all the three nodes. A just needs to alternatively transmit data to B and C to avoid collision.

The reason why MMAC causes unnecessary delay in the above cases is that only pairs of nodes negotiate common channels in each interval, and thus each packet can be transmitted at most one hop away in one interval. If we enable more than two nodes to negotiate a common channel, the transmission delay can be dramatically reduced. For example, in both cases of Fig.2, if A, B, and C can negotiate a common channel together, then all the transmissions can be completed in one time interval. Next, we present the design of Adaptive Dynamic Channel Allocation protocol (ADCA).

ADCA uses the similar framework with MMAC. It divides time into fixed length intervals. Each interval is further split into control interval and data interval. Let T be the interval length, and T_c, T_d be the length of control interval and data interval respectively (Obviously, we have $T = T_c + T_d$). In the control interval, all nodes switch to the same default channel and negotiate channels. In the data interval, the nodes working on the same channel transmit and receive data among each other. In MMAC, T is set to $100ms$, and T_c is set to $20ms$, which is long enough for nodes to negotiate channels when network traffic is saturated. Our protocol uses the same

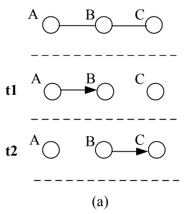


Fig. 2: Unnecessary Delay of MMAC when Traffic is below Saturation

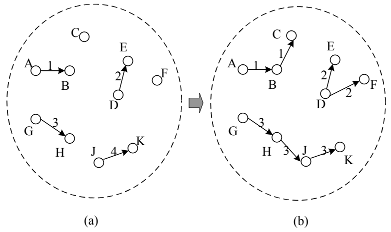


Fig. 3: Adaptive Dynamic Channel Allocation

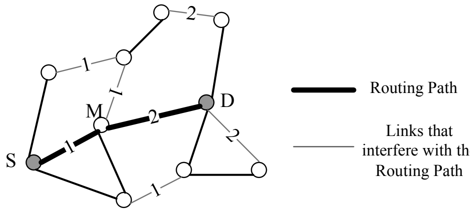


Fig. 4: Effect of Inter-flow Interference

parameter settings (T and T_c), but is different in the channel allocation scheme during control interval.

In ADCA, each dynamic interface maintains multiple queues in the link layer with one queue for each neighbor. The data to be sent to each neighbor are buffered in the corresponding queue. The first step of channel negotiation in ADCA is similar to MMAC. For each dynamic interface, if it has data to transmit, it selects a neighbor that it wants to communicate with and tries to negotiate a common channel with the neighbor. There are many criteria for selecting neighbors. If throughput is the only consideration, we may select the neighbor with the longest queue. However, this strategy may cause starvation. Therefore, we augment it with some fairness considerations, that is, we evaluate a neighbor's priority by considering both its queue length and how long the queue has not been served. As a result, during this step, pairs of nodes have negotiated common channels with each other such as the example in Fig.3(a).

Different from MMAC, ADCA enables further channel negotiation among nodes. The example in Fig.3 illustrates how our protocol works. Assume the network traffic is below saturation. In Fig.3(a), pairs of nodes negotiate channels according to MMAC. Then further channel negotiations are performed as illustrated in Fig.3(b). There are three cases illustrated in the figure. 1) A has some data to send to C through B. A and B got the right to transmit data on a common channel, while B and C did not. In this case, B can further negotiate with C so that C works on the same channel with A and B. 2) D has some data to both E and F. D and E got the right to transmit data on a common channel, while D and F did not. In this case, D can further negotiate with F so that F works on the same channel with D and E. 3) G has some data to J through H. G and H got the right to transmit data on a common channel, while J got the right to transmit data to K on a common channel. In this case, H can negotiate with J so that G, H, J, and K work on the same channel.

Compared with MMAC, ADCA can negotiate common channels among more than two nodes in each interval when network traffic is not saturated. As a result, ADCA has the potential to reduce packet delay while satisfying the imposing traffic. When the traffic is near saturated, ADCA will behave similar to MMAC. MMAC is only optimized for maximizing the network capacity. In contrast, ADCA optimizes for both throughput and delay with regard to the imposing traffic load. Therefore, ADCA is adaptive to the network traffic.

IV. INTERFERENCE AND CONGESTION AWARE ROUTING

The previous routing metrics proposed in wireless mesh networks include hop count, RTT, ETX, ETT, WCETT [14]. These metrics aim at finding a good quality path for a single flow. In this paper, our goal is to maximize the total throughput in the hybrid network with both static links and dynamic links. We want the routes of different flows to be selected efficiently such that the channel usages are balanced at each node and thereby avoiding congestion in the network. We propose an interference and congestion aware routing metric as follows.

As shown in Fig.4, assume the routing path of a flow from S to D is illustrated in bold lines, and the rate of the flow is r. All the wireless links that interfere with the routing path are plotted in grey lines. Now, let us consider the bandwidth that the transmission of the flow will consume in the network. The flow will consume bandwidth of r on link SM. However, there are three other links that interfere with link SM. As interfering links cannot be active at the same time, we may think that the flow is also consuming the same bandwidth in the other interfering links. It is a similar case with link MD. Therefore, although the flow is routed through two wireless links, it is actually consuming bandwidth of multiple links in the network.

To describe it formally, let P be the routing path of a flow with rate r. For each link $l \in P$, let $IE(l)$ be the set of links that interfere with l (assume $IE(l)$ also contains l). Let $ETX(l)$ be the expected transmission count of link l. The total bandwidth that is consumed by routing the flow can be calculated by the following formula.

$$B_{total} = \sum_{l \in P} \mid IE(l) \mid * ETX(l) * r$$

The B_{total} metric reflects the resource consumed by routing the flow through the network. In order to support as many flows in the network as possible, the routing of each flow cannot be too selfish. This metric can help balance the channel usage better than the previous metrics. In order to make the routing better at avoiding congested links, we make some further modifications on B_{total} in the following.

We categorize each link into three states: Congested, Median, and Low. "Congested" means the channel on which the link is working is highly congested, and therefore it is not desirable to route additional traffic through this link. “Median” means the channel used by the link has a considerable amount of load, but is still able to route additional traffic. “Low” means the channel has very light load, and thus the link is preferred to be used to route additional traffic. Therefore, we can assign different weights $w1$, $w2$, $w3$ to three different states, which reflects the cost of adding additional traffic to the link in different states. Obviously, they should satisfy $w1 > w2 > w3$. By inducing the weights into B_{total}, the cost of routing a flow at rate r along a path P can be calculated as follows.

$$Cost_{total} = \sum_{l \in P} [ETX(l) * r * \sum_{e \in IE(l)} w(e)]$$

Therefore, we can define a cost metric of each link as $C(l) = ETX(l) * r * \sum_{e \in IE(l)} w(e)$ and then use source routing to find a minimum cost path for each flow, $P^* = argmin \sum_{l \in P} C(l)$.

There remains two issues in our routing algorithm, the determination of links states and interfering links.

The link states can be inferred from the queue length. In the hybrid architecture, each static interface has one queue while each dynamic interface maintains multiple queues, one for each neighbor. The state of each link, whether static or dynamic, can be inferred from the corresponding queues of the two end interfaces. The longer the average queue length, the more likely is the link congested.

Unlike the static link whose channel is fixed, a dynamic link may work on different channels at different intervals. Thus, if there is a dynamic link in a pair of links, we cannot deterministically say whether they will interfere with each other or not. An alternative way is to estimate their probability of interference.

For each dynamic link, we maintain statistics of its channel usage history, in the form of $\{p_i | i = 1, 2, ..., C\}$, where p_i is the fraction of time that the link worked on channel i. Note that this representation is also applicable to static links. If the static link works on channel c, then $p_c = 1$ and $p_i = 0 (i \neq c)$ in its representation. Given two links, l_1 and l_2, whose channel usage is represented by $\{p_i^{l_1}\}$ and $\{p_i^{l_2}\}$, we can predict their probability of interference using the following formula.

$$P_{if}(l_1, l_2) = \sum_{i \in \{1,2,...,C\}} p_i^{l_1} * p_i^{l_2}$$

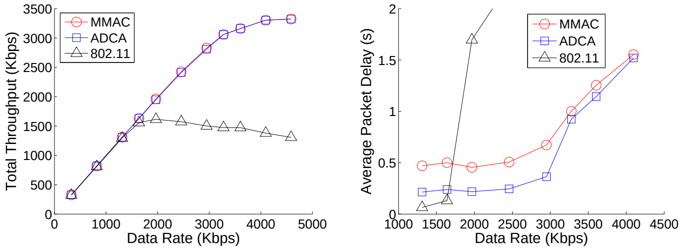
Therefore, the cost formula can be modified as:

$$Cost_{total} = \sum_{l \in P} [ETX(l) * r * \sum_{e \in IE(l)} (w(e) * P_{if}(e, l))]$$

The channel usage array is easy to maintain in each node, because dynamic links change channels in a synchronous fashion, that is, interval by interval. This information can be updated to routing agents periodically.

V. PERFORMANCE EVALUATION

We performed simulations in NS-2.31. The Hyacinth extension [15] was used to support multiple channels and multiple interfaces per node in the simulator. We made further extensions to support dynamic channel switching on each interface. In this section, we first evaluate ADCA by comparing it with MMAC. After that, we evaluate the hybrid wireless mesh network and compare it with pure static architectures. In all the simulations, the bit rate of each interface is set to $11Mbps$. The radio transmission range is 250 meters, and interference range is 500 meters.



(a) Network Throughput

(b) Packet Delay

Fig. 5: Performance of ADCA under 3 Channels

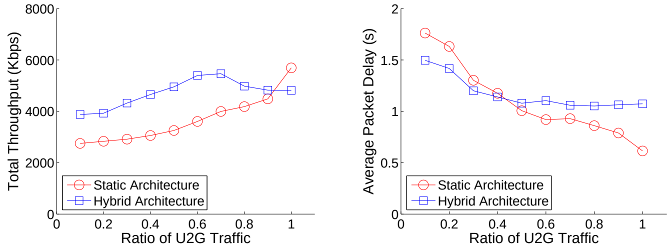
A. *Evaluation of Adaptive Dynamic Channel Allocation*

We first perform the simulation on a random topology of 50 nodes in a $1200m \times 1200m$ area. Each node has at least 2 neighbors and at most 8 neighbors. The average degree of the nodes is 5.4. To compare with MMAC, each node has only one interface, which can switch channels dynamically. We generate 25 UDP flows with the same data rate in the network, each with random source and random destination. The packet size in each flow is set to 512 bytes.

We analyze three protocols: 1) 802.11, a single-channel MAC protocol; 2) MMAC, a multi-channel MAC protocol; 3) ADCA, an adaptive multi-channel MAC protocol. For both MMAC and ADCA, we use 3 orthogonal channels. By using the same parameter in [11], we set the time interval to $100ms$, and the control interval to $20ms$. These three protocols are compared with regard to throughput and delay in Fig.5.

In Fig.5(a), with the increasing data rate, the network is gradually becoming saturated. We can observe that ADCA achieves the same throughput with MMAC. The maximum throughput of ADCA and MMAC is over twice of 802.11 because of the utilization of multiple orthogonal channels.

Fig.5(b) illustrates the average packet delay with varying data rate. We can observe that when the traffic load is below saturation ($1600Kbps$) of single-channel MAC, 802.11 has the lowest delay. However, with the increasing traffic load, 802.11 suffers a sharp increase in delay because packets are experiencing long waiting time in queues. On the other hand, as multi-channel MAC achieves higher capacity, the delay is much lower than 802.11 when the traffic load is over $1600Kbps$. In comparison with MMAC, ADCA dramatically decreases the packet delay when the traffic load is below $3000Kbps$. Particularly, in the interval of $[1800Kbps, 3000Kbps]$, ADCA achieves dramatically lower delay among all three protocols. The larger delay of 802.11 is because of the long waiting time in queue, while the larger delay of MMAC is because it only



(a) Network Throughput

(b) Packet Delay

Fig. 6: 50 nodes with 6 channels under Mixed Traffic

allows packets to be delivered one hop in each time interval. At loads larger than $3000Kbps$, the improvement of ADCA over MMAC is less dramatic, because the network is getting saturated and ADCA does not further negotiate channels. From the simulation results, we can conclude that ADCA is able to achieve lower delay than MMAC without impacting the network throughput.

B. *Evaluation of Hybrid WMN*

In this section, we evaluate the throughput of the hybrid network, and compare it with the pure static architecture. We consider a wireless mesh network with 50 nodes randomly deployed in a $1200m \times 1200m$ area. There is one node designated as the gateway node in the middle area. Each node has 3 interfaces. In the pure static network, all the interfaces of each node work with static channels, while in the hybrid network, 2 interfaces of each node work with static channels and the other interface works with dynamic channels. In the simulation, we use the interference and congestion aware routing algorithm on the hybrid architecture and use ETX routing on the static architecture.

We consider two types of traffic in the network: 1) $U2G$, connections between end-users (non-gateway nodes) and the gateway; 2) $P2P$, connections between end-users. In both types, end-user nodes are selected randomly. In our experiment, we generate 50 random flows, each with the same data rate, and vary the ratio of the number of $U2G$ flows over all flows ($U2G$ and $P2P$). The maximum network throughput is measured by the total throughput of all flows when the network is getting saturated by the flows.

Fig.6 illustrates the simulation results with 6 orthogonal channels. As shown in Fig.6(a), the static architecture works best when the ratio is 1 (there is only $U2G$ traffic). With the decreasing ratio, the throughput of the static network decreases dramatically, because it is not able to provide good routing paths for $P2P$ traffic in most cases. In contrast, there is no dramatic throughput degradation for hybrid network with the changing ratio. This is because the network connectivity is well-maintained in the hybrid architecture, and dynamic links can be used to construct good routing paths with static links when necessary. We also observe that the hybrid architecture always outperforms the static architecture except when the ratio is close to 1, because the channel allocation of the static network is especially optimized for this particular case. Fig.6(b) shows the average packet delay. When the ratio is over 0.4, the hybrid architecture has higher delay, which is mainly caused by dynamic channel switching. When the $P2P$ traffic becomes dominant, the static architecture tends to have higher delay because of the increased waiting time in queues.

VI. CONCLUSION

In this paper, we have proposed a hybrid wireless mesh network architecture, where each mesh node has both static and dynamic interfaces. We have made two contributions. First, we proposed an adaptive dynamic channel allocation protocol (ADCA) to be used on dynamic interfaces. Compared with MMAC, ADCA reduces the packet delivery delay without degrading the network throughput. In addition, we proposed an interference and congestion aware routing algorithm in the hybrid network, which balances the channel usage in the network, thereby increasing the network throughput. Our simulation results have shown that, compared to the pure static architecture, the hybrid architecture is more adaptive to the changing traffic without dramatic increase in overhead.

VII. ACKNOWLEDGEMENT

This work was supported in part by the US National Science Foundation under grants CCF-0514078, CNS-0551464, and CNS-0721441.

REFERENCES

[1] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," in *Computer Networks*, 2004.
[2] P. Gupta and P. R. Kumar, "The capacity of wireless networks," in *IEEE Transactions on Infomation Theory*, 2000.
[3] J. Padhye, S. Agarwal, V. N. Padmanabhan, L. Qiu, A. Rao, and B. Zill, "Estimation of link interference in static multi-hop wireless networks," in *Internet Measurement Conference*, 2005.
[4] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *MobiCom*, 2003.
[5] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," in *MC2R*, 2004.
[6] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *MobiCom*, 2005.
[7] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *MobiCom*, 2005.
[8] A. Raniwala and T. Chiueh, "Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network," in *INFOCOM*, 2005.
[9] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and qos routing in multi-channel wireless mesh networks," in *MobiHoc*, 2005.
[10] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel mac protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *Proceedings of the 2000 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN)*, 2000.
[11] J. So and N. Vaidya, "Multi-channel mac for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *MobiHoc*, 2004.
[12] 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.
[13] W.-H. Tam and Y.-C. Tseng, "Joint multi-channel link layer and multi-path routing design for wireless mesh networks," in *INFOCOM*, 2007.
[14] R. Draves, J. Padhye and B. Zill, "Routing in multi-radio multi-hop wirelss mesh networks," in *MobiCom*, 2004.
[15] "Hyacinth: An ieee 802.11-based multi-channel wireless mesh network, http://www.ecsl.cs.sunysb.edu/multichannel/."