Routing and Interface Assignment in Multi-Channel Multi-Interface Wireless Networks*

Technical Report, October 2004

Pradeep Kyasanur
Dept. of Computer Science, and
Coordinated Science Laboratory,
University of Illinois at Urbana-Champaign
Email: kyasanur@uiuc.edu

Nitin H. Vaidya
Dept. of Electrical and Computer Engineering, and
Coordinated Science Laboratory,
University of Illinois at Urbana-Champaign
Email: nhv@uiuc.edu

***Abstract*— Multiple channels are available for use in IEEE 802.11. Multiple channels can increase the available network capacity, but require new protocols to exploit the available capacity. This paper studies the problem of improving the capacity of multi-channel wireless networks by using multiple interfaces. We consider the scenario when multiple interfaces are available, but the number of available interfaces is lesser than the number of available channels. We propose algorithms for assigning interfaces to channels that do not require modifications to IEEE 802.11. We also propose a routing protocol that is suitable for use with the proposed interface assignment strategy.**

I. INTRODUCTION

IEEE 802.11 [1] is a widely used technology for wireless local area networks. IEEE 802.11 offers multiple non-overlapping channels that are separated in frequency. For example, IEEE 802.11b offers 3 non-overlapping channels, while IEEE 802.11a offers 12 non-overlapping channels. Multiple channels have been exploited in infrastructure-based networks by assigning different channels to adjacent access points, thereby minimizing interference between access points. However, typical multi-hop wireless network configurations require a single common channel to be used by all nodes to ensure network connectivity. Our goal in this paper is to utilize the multiple channels in multi-hop wireless networks.

Inexpensive commodity IEEE 802.11 hardware has accelerated the use of wireless local area networks. This trend of reducing hardware costs is expected to continue [2], and it is already feasible to equip nodes with multiple 802.11 interfaces. However, it is still expensive to equip a node with *one interface for each channel* (recall that IEEE 802.11a has 12 non-overlapping channels) . Many IEEE 802.11 interfaces can be *switched* from one channel to another, albeit at the cost of a switching delay, thereby allowing an interface to access multiple channels. In this paper, we study the multi-channel problem when the number of interfaces is lesser than the number of channels, and address the following questions: *What is a suitable strategy for assigning interfaces to channels? What is the impact of interface assignment on the routing protocol?*

In this paper, we propose an interface assignment strategy that keeps one interface fixed and switches the other interfaces. The interface assignment strategy ensures that any two nodes within communication range of each other can communicate without requiring specialized coordination algorithms. We then propose routing strategies that are well-suited for use with the proposed interface assignment strategy. Past work on multi-channel, multi-interface wireless networks has mostly focused on MAC protocols, while we primarily focus on the routing and interface assignment problem on top of existing IEEE 802.11 MAC protocol.

The rest of the paper is organized as follows. Section II presents the related work. Section III motivates the benefits of multiple interfaces, and the need for specialized routing protocols for multi-channel, multi-interface networks. Section IV describes the interface assignment strategy, and Section V describes the routing protocol. Section VI has a discussion on other issues with multi-channel, multi-interface networks and we conclude in Section VII.

II. RELATED WORK

Several researchers have proposed MAC protocols based on IEEE 802.11 for utilizing multiple channels. Nasipuri et al. [3], [4], and Jain et al. [5] propose a class of protocols where all nodes have an interface on each channel. The protocols differ in the metric used to choose a channel for communication between a pair of nodes. The metric may simply be to use an idle channel [3], or the signal power observed at the sender [4], or the received signal power at the receiver [5]. These protocols are expensive to implement as an interface is needed for each channel.

Wu et al. [6] propose a MAC layer solution that requires two interfaces. One interface is assigned to a common channel for control purposes, and the second interface is switched between the remaining channels and used for data exchange. RTS/CTS packets (as in IEEE 802.11) are exchanged on the control channel, and the exchange also determines the appropriate data channel to be used for subsequent DATA/ACK exchange. Hung et al. [7] propose a similar two-interface solution that uses a channel load-aware algorithm to choose the appropriate data channel to be used for DATA/ACK exchange. While both proposals require only two interfaces to support any number

---

*This research was supported in part by NSF grant ANI-0125859 and a Vodafone Graduate Fellowship.

of channels, the common control channel may become a bottleneck to performance. Since RTS/CTS exchange precedes *each data transmission*, the approach does not scale when the number of data channels is large (e.g., 12 channels with 802.11a).

So et al. [8] propose a MAC solution for multiple channels that uses a single interface. Nodes periodically switch to a common channel, and stay on the common channel for a fixed negotiation duration when the channel to be used for later data transmission is decided. At the end of the fixed negotiation phase, nodes switch to the chosen channel for data communication. This proposal requires a single interface and can be implemented by extending IEEE 802.11 Power Save Mode. However, the solution requires tight synchronization among nodes, which is still a hard problem for multi-hop networks.

All the multi-channel MAC proposals described above require changes to IEEE 802.11, and therefore cannot be deployed by using commodity hardware. In contrast, our proposal can be implemented with standard 802.11 interfaces.

Adya et al. [9] propose a link-layer solution for striping data over multiple interfaces. The proposal does not use interface switching, and for full utilization of available channels, an interface is necessary for each channel. Hence, this proposal is expensive to implement when large number of channels are available.

Bahl et al. [10] propose SSCH, a link-layer solution that uses a single interface, and can run over unmodified IEEE 802.11 MAC. Nodes implementing SSCH use a pseudo-random sequence, driven by a set of seeds, to decide which channel to switch the interface to every time slot. The pseudo-random sequence used by any two nodes is guaranteed to overlap periodically, thereby ensuring any two nodes within communication range can communicate with each other. Frequently communicating nodes can partially synchronize their seeds to increase the overlap frequency. While a single interface is sufficient for SSCH operation, it may introduce significant delay with multi-hop communication, as packets may be delayed at each hop if the subsequent hop node is on a different channel.

Draves et al. [11] propose WCETT, a new metric for routing in multi-channel networks. The metric is used with LQSR, a source routing protocol, and ensures "high-quality" routes are selected. In contrast to our work, LQSR does not use interface switching, and is not designed for the scenario when number of available interfaces is less than the number of available channels.

Shacham et al. [12] propose a architecture for multi-channel networks that uses a single interface. Each node has a default channel for receiving data. A node with a packet to transmit has to switch to the channel of the receiver before transmitting data. However, the proposal does not consider the impact of switching delay. Further, the routes used in the architecture may not utilize multiple channels.

So et al. [13] propose a routing protocol for multi-channel networks that uses a single interface at each node. We propose to use multiple interfaces, which may offer better performance than a single interface solution.

Raniwala et al. [14], [15] propose routing and interface assignment algorithms for mesh networks. Their goal is similar to our work in addressing the scenario where the number of available interfaces is less than the number of available channels. However their approach is different in the following key aspects. Raniwala's protocol assumes traffic load between all nodes are known, and centralized algorithms are used to derive an assignment of interfaces to channels and for route computation. In contrast, we do not make any assumptions on the traffic characteristics, and our algorithms are completely distributed. In addition, the routes selected by their approach may be significantly longer than our proposal as interface switching is not used.

In the context of wired networks, Marsan et al. [16] have studied the performance of multichannel CSMA/CD MAC protocols, and shown that significant reduction in delay average and variance is possible even when the number of interfaces is less than the number of channels. The goal of our work is to answer a similar question with multi-channel CSMA/CA based wireless networks. We intend to study the impact of routing strategies as well.

III. MOTIVATION

In this section, we first motivate the benefits of using a multi-interface solution for exploiting multiple channels. We then identify the need for specialized routing protocols for multi-channel, multi-interface networks.

A. Benefits of using multiple interfaces

We define "interface" to be a network interface card equipped with a half-duplex radio transceiver, e.g., a commodity 802.11 wireless card. In most multi-hop networks, a single channel is used, and therefore a single interface suffices. However, when multiple channels are available, having more than one interface is beneficial.

As noted while describing related work, there are single interface approaches ([8], [10], [13]) for exploiting multiple channels. When using a single interface, if the interfaces of two nodes are on different channels, then they cannot communicate. For reducing synchronization requirements and overheads, each interface has to stay on a channel for many packet transmission durations (100ms in [8] and 10ms in [10]). As a result, when packets are traversing multi-hop paths, packets may be delayed at each hop, unless the next hop is on the same channel as well. Thus, when a single interface is used, there is an increase in the end-to-end latency if different hops traversed are on different channels. Otherwise, if most hops are on the same channel, transmissions on consecutive hops interfere, reducing the maximum capacity. In either case, TCP throughput is significantly affected.

When at least two interfaces are available, we propose keeping one interface permanently assigned to a channel to greatly simplify coordination, while switching the second interface (based on traffic requirements) to avoid delaying a packet at each hop. We defer discussion of the proposed approaches till later in the paper, but multiple interfaces are required to derive both simplicity in coordination and minimal delays.

A second benefit is the ability to receive and transmit data in parallel. Half-duplex wireless interfaces cannot simultaneously transmit and receive data. However, when multiple (say two) interfaces and multiple channels are available, while one interface is receiving data on one channel, the second interface can simultaneously transmit data on a different channel. In many cases, this can double the maximum throughput achievable on a multi-hop route. Our proposed architecture exploits this benefit of using multiple interfaces as well.

B. Issues with interface switching

The ability to switch an interface from one channel to another is a key property we exploit to utilize all the available channels, even when the number of interfaces available is significantly lesser than the number of available channels. We assume that channels are separated in frequency, and switching an interface requires changing the frequency of operation. Switching an interface from one channel to another incurs some delay D which may be non-negligible. In the current literature, estimates for D (for switching between channels on the same frequency band) with commodity IEEE 802.11 hardware are in the range of a few milliseconds [17] to a few hundred microseconds [18]. It is expected that with improving technology, the switching delay will reduce to a few tens of microseconds [10]. Protocols that utilize interface switching need to be flexible enough to accommodate a range of switching delays. The routing protocol may have to account for the switching cost while selecting routes.

Interface switching is possible across different frequency bands as well. For example, wireless cards are currently available that support both IEEE 802.11a (operates on 5 GHz band) and IEEE 802.11b (operates on 2.4 GHz band), and can switch between the two bands. However, with the currently available hardware, switching across bands incurs a large delay, but the switching delay is expected to reduce in the future. The architecture presented in this paper allows for the utilization of channels on the same band as well as channels on different bands.

C. Need for specialized routing protocols

Existing routing protocols for multi-hop networks such as DSR [19] and AODV [20] support multiple interfaces at each node. However, those protocols typically select shortest-hop routes, which may not be suitable for multi-channel networks, as was noted in [11]. In addition, if route selection does not consider the interface switching cost, then the chosen routes may require frequent channel switching, degrading network performance. Thus, there is a need for customized protocols for multi-interface, multi-channel networks.

Figure 1 illustrates a scenario that highlights the need for specialized routing protocols for multi-channel networks. In the figure, node A is communicating with node D using route A-C-D. Node E wishes to communicate with node F, and either of B or C can be used as the intermediate node. Assume all nodes have a single interface, and assume C and B can relay at most w bytes per second. If node C is chosen as the intermediate node, then node C has to forward data along both routes A-C-D and E-C-F, and the throughput received by each flow is at most $w/2$. On the other hand if node B is chosen as the intermediate node, then both routes A-C-D and E-B-F can be simultaneously used (assuming channels used on routes A-C-D and E-B-F can be chosen to be orthogonal), and each flow receives a rate of w. Although this example assumed each node had a single interface, similar issues arise even when multiple interfaces are available.

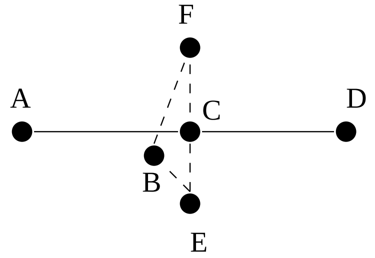

Fig. 1. Impact of route selection on effective utilization of multiple channels

The above scenario highlights the need for the routing protocol to appropriately *distribute* routes among nodes in the neighborhood. In the case of single channel networks, the throughput obtained is the same whether B or C is chosen as the intermediate node. When a single channel is available, and say, when C is transmitting a packet along route A-C-D, B cannot transmit a packet even if it is chosen as the intermediate node (as the common channel is busy). Consequently, routing protocols designed for single channel networks do not need to distribute routes within a “neighborhood”. However, to exploit the benefit of multiple channels, it is important for a routing protocol to ensure routes are carefully distributed in the network.

IV. INTERFACE ASSIGNMENT

In this section, we identify the different interface assignment strategies possible. We then describe our proposal and discuss issues involved.

A. Classification of interface assignment strategies

Interface assignment strategies can be classified into static, dynamic, and hybrid strategies.

1. Static Assignment: Static assignment strategies assign each interface to a channel either permanently, or for “long intervals” of time where “long interval” is defined relative to the interface switching time. For example, [11], [14] use static interface assignment. Static assignment can be further classified into two types:

1) Common channel approach: In this approach, interfaces of all nodes are assigned to a common set of channels (e.g. [11]). For example, if two interfaces are used at each node, then the two interfaces are assigned to the same two channels at every node. The benefit of this approach is that the connectivity of the network is the same as that of a single channel approach. Note that the scenario where a single channel and a single interface is used is a special case of the static, common channel assignment strategy.
2) Varying channel approach: In this approach, interfaces of different nodes may be assigned to a different set of channels (e.g. [14]). With this approach, there is a possibility that the length of the routes between nodes

may increase. Also, unless the interface assignment is done carefully, network partitions may arise.

Static assignment strategies are well-suited for use when the interface switching delay is large. In addition, if the number of available interfaces is equal to the number of available channels, interface assignment problem becomes trivial. Static assignment strategies do not require special coordination among nodes (except perhaps to assign interfaces over long intervals of time) for data communication. With static assignment, nodes that share a channel on one of their interfaces can directly communicate with each other, while others cannot. Thus, the effect of static channel assignment is to control the network topology by deciding which nodes can communicate with each other.

2. Dynamic Assignment: Dynamic assignment strategies allow any interface to be assigned to any channel, and interfaces can frequently switch from one channel to another. In this setting, two nodes that need to communicate with each other need a coordination mechanism to ensure they are on a common channel at some point of time. For example, the coordination mechanism may require all nodes to visit a common "rendezvous" channel periodically (e.g. [8]), or require other mechanisms such as the use of pseudo-random sequences (e.g. [10]), etc. The benefit of dynamic assignment is the ability to switch an interface to any channel, thereby offering the potential to cover many channels with few interfaces. The key challenge with dynamic switching strategies is to coordinate the decisions of when to switch interfaces as well as what channel to switch the interfaces to, among the nodes in the network.

3. Hybrid Assignment: Hybrid assignment strategies combine static and dynamic assignment strategies by applying a static assignment for some interfaces and a dynamic assignment for other interfaces. Hybrid strategies can be further classified based on whether the interfaces that apply static assignment use a common channel approach, or a varying channel approach. An example of hybrid assignment with common channel at the MAC layer is [6], which assigns one interface of each node statically to a common "control" channel, and other interface can be dynamically switched among other "data" channels. We propose to use a hybrid channel assignment strategy with varying channel assignment. Hybrid assignment strategies are attractive as they allow simplified coordination algorithms supported by static assignment while retaining the flexibility of dynamic assignment.

B. Interface Assignment Protocol

We assume that there are M interfaces available at each node, where the value of M may be different for different nodes. Some K of the M interfaces at each node are statically assigned to K channels, and we designate these interfaces as "fixed interfaces", and the corresponding channels as "fixed channels". The other $M-K$ interfaces, designated as "switchable interfaces", are dynamically assigned to any of the remaining $M-K$ channels, based on data traffic. Different nodes may assign their K interfaces to a different set of K channels. It is also possible for each node to use a different value of K, and it is also possible to vary K with time. To simplify rest of the discussion, we assume $M = 2, K = 1$ for all nodes, i.e., there is one fixed, and one switchable interface (although the proposed protocol is applicable to any values of M and K).

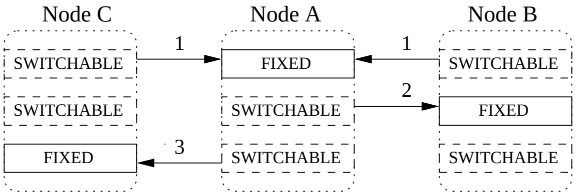

Fig. 2. Example of switching protocol operation ($M = 2, K = 1$)

We illustrate the use of fixed and switchable interface with the example topology in Figure 2. Assume node A wishes to exchange data with nodes B and C. Further, assume that the fixed interface of node A is on channel 1, while the fixed interface of nodes B and C are on channels 2 and 3 respectively. When A has to send a packet to B, A switches its switchable interface to channel 2 and transmits the packet. Since B is always *listening* to channel 2 with its fixed interface, B can receive the transmission of A. Now if B has to send a packet back to A, B switches its switchable interface to channel 1 and transmits the packet. Since A is listening to channel 1 with its fixed interface, the packet from B can be received. Similarly, if A has to subsequently send a packet to C, it switches to channel 3 and sends the packet. Note that B and C can at any time send a packet to A on channel 1. Thus, there is no need for coordination among A, B, and C on *when to schedule transmissions.*

1) Supporting Broadcasts:

In wireless networks, all packets transmitted on a channel can be received by all neighboring nodes listening to that channel. In single-channel networks this property is used to support efficient neighborhood broadcast, which is used by on-demand routing protocols in the route discovery process. However, a similar broadcast property is not inherently available when multiple channels are used, as nodes in a neighborhood may be listening to different channels. For achieving an equivalent broadcast property when using multiple channels, the broadcast packet has to be separately transmitted on all channels. Thus, broadcast can be more expensive than in single channel networks. Furthermore, the broadcast packets on different channels may be sent at slightly different times (as the switchable interface has to be switched through all channels). Thus, nodes with fixed interfaces on different channels may receive the broadcast at different times. Routing protocols may have to account for the modified broadcast semantics.

An enhancement is possible when the number of available channels is large, and at least three interfaces are available. One channel can be set apart in the whole network for broadcast purposes, and each node can assign one interface permanently to the broadcast channel (e.g., when $M = 3, K = 2$). All broadcast transmissions can be sent on the special broadcast channel. The use of a broadcast channel differs from existing MAC proposals that use a common control channel, as the control channel is used for every unicast/broadcast transmission, while the broadcast channel is used infrequently for broadcast transmissions only.

2) Fixed interface assignment and discovery: The use of fixed interfaces raises two questions. How does a node X decide what channel to assign to the fixed interface? How do neighbors of node X know about the fixed channel used by X? We propose two approaches for solving this problem.

In the first approach, each node uses some well-known function f (e.g., f can be a function which generates a hash based on its input) of its node identifier to select the channel to assign to the fixed interface. Neighbors of a node X can use the same function f to compute the fixed channel used by X. This approach is simple, but there is a possibility that some channels in a neighborhood will not be used by any node.

In the second approach, initially every node selects a random channel as the fixed channel. Each node periodically broadcasts a "Hello" packet informing its neighbors of its fixed channel. Based on the received "Hello" packets, nodes may (with some probability, to avoid oscillations) choose to set their fixed channel to an unused or a lightly loaded channel. "Hello" packets may be anyway needed in the face of mobility and this mechanism is expected to be inexpensive. This approach ensures that there is a high probability that all channels are used. One improvement is to consider the channel quality, in addition to the information received from neighbors, when deciding on the choice of a fixed channel.

3) Switchable interface management: The switchable interface on a node X is used to transmit data whenever the fixed channel of the destination is different from the fixed channel of X. One issue to be resolved is how frequently to switch channels. For example, consider a stream of packets at a node X where the even-numbered packets are to destination A, and the odd numbered packets are to destination B, with A and B on different channels. Thus, a policy is needed to decide when to switch an interface, and what channel to switch the interface to? One possibility is to alternately switch between channels for *each packet*. However, such frequent switching may be very expensive when the switching delay is large. Another possibility is to switch over longer intervals of time, thereby amortizing the cost of switching among multiple packets.

Based on the above discussion, we propose an architecture, depicted in Figure 3. Each channel has a separate queue. The switchable interface services at most k packets on one channel, before switching to another channel (only if there are packets for some other channel). In addition, the switchable interface stays on a channel for at most t seconds, before switching to another channel (again, switching happens only if there are packets for some other channel). The two conditions in conjunction ensure that the extra latency introduced by the switching protocol is bounded by t, while the switching cost is amortized among up to k packets. The parameters k and t can be suitably set to trade-off latency with performance.

The switching algorithm may need to support fairness. For example, in the architecture described above, when switching an interface, we can support fairness by switching to a channel having the oldest data packet in its queue.

4) Key benefits of the proposed interface assignment strategy: The proposed switching architecture and protocols have many useful properties.

1) The architecture can be built over existing MAC protocols, such as IEEE 802.11.

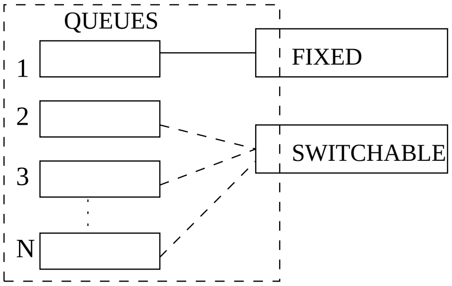

Fig. 3. Example architecture with N channels and two interfaces ($M = 2, K = 1$)

2) The sender and receiver nodes do not need to synchronize for channel switching. In addition, there is no need for specialized coordination algorithms to guarantee that the sender and receiver are on the same channel.
3) By carefully balancing the assignment of fixed interface over the available channels, the number of contending transmissions in a neighborhood significantly reduces.
4) The protocol can easily scale if the number of available channels increases.

V. ROUTING STRATEGY

Various routing protocols have been proposed for multi-hop wireless networks. Most of the commonly used routing protocols such as DSR and AODV select shortest-path routes. However, the shortest path metric may not be suitable for multi-channel, multi-interface networks as it does not exploit the available channel diversity. For example, the shortest path metric does not distinguish between a route with x hops, each on a different channel (resulting in low contention), and another route with all x hops on a single channel (resulting in high contention). Further, the shortest path metric does not account for the impact of interface switching. In this section, we first discuss techniques to quantify the cost of interface switching and channel diversity, and then propose routing heuristics that incorporate the impact of switching cost and channel diversity.

A. Cost of interface switching

Switching delay impacts a route only if a node is forwarding data along multiple routes. If all data through a node is along one route, then after the interface is initially switched on to the desired channel, no further switching is necessary. More formally, switching delay impacts a node only if the number of distinct, non-fixed channels a node uses is more than the number of available interfaces. We designate nodes impacted by the switching delay as "interface bottlenecked" nodes.

The cost of switching for a channel is a combination of the switching delay and how frequently an interface is switched to that channel. For example, when we use the strategy (described in Section IV-B.3) of switching once only in k packets, switching cost is amortized over k packets. So, if the switching delay is D seconds for each switch, we can assign the switching cost to be D/k for each packet. The cost of interface switching along a route may be measured in terms of the number of "interface bottlenecked" nodes along the route, or in terms of the total switching cost along each node in the route.

B. Measuring channel diversity

The availability of multiple interfaces enables a node to transmit and receive data in parallel, provided different channels are used for transmission and reception. If each node along a route chooses different channels for reception and transmission, higher throughput can be achieved. More formally, if a node along a route can interfere with r other nodes along the route, then for higher performance, the channels used by the node (for receiving data) and the r interfering nodes must be different. We define "diversity cost" to be the cost incurred by a node on a route due to interference with other nodes along the same route.

One way of measuring diversity cost (max-interference method) that we propose is as follows. If a node X along a route has i other interfering nodes receiving on the same channel as X, then the diversity cost of X is defined to be i. Since the end-to-end performance is impacted by the performance of a bottleneck link along the route, the diversity cost of the whole route is defined as the *maximum diversity cost of any node along the route.*

Diversity cost can also be measured using the ETT metric, defined in [11]. Expected Transmission Time (ETT) is the average transmission time for packet exchange between two nodes, after accounting for retransmissions. In [11], the sum of ETT on all hops of a route that use a common channel is defined to be the diversity cost for that channel. The diversity cost of the whole route is defined as the *maximum diversity cost of any channel along the route.*

C. Routing Heuristics

The routing protocol has to select routes which have low switching cost as well as low diversity cost, for maximizing the throughput obtained. In addition, the routing protocol has to account for global resource usage as well (e.g., total number of hops traversed along a route), to avoid inefficient resource utilization. Thus, we can compute the total cost of a route as the weighted combination of the switching cost, the diversity cost, and the global resource usage cost. It is part of our ongoing work to study the appropriate weights to be used, and the trade-offs involved with different weights. Different routing heuristics can be developed by using different approaches to measure each cost, and by using different approaches for combining the costs. To illustrate the possibilities, we propose two different metrics that can be used for routing.

1) Enhanced shortest path metric: This metric measures the switching cost as the number of interface bottlenecked links, the diversity cost using the maximum interference method, and the global resource usage cost as the total number of hops on the route. This metric is simple to use and can be computed as part of the route discovery process itself.
2) Enhanced WCETT metric: Draves et al. proposed a metric called "WCETT" [11] based on ETT. But WCETT does not account for switching cost, and hence we propose the "enhanced WCETT" metric. Enhanced WCETT metric measures the switching cost as the sum of switching delays along the route, the diversity cost using the ETT metric, and the global resource usage cost as the sum of ETT values along the path. WCETT metric has been shown to perform well (when interface switching is not used) in multi-channel scenarios, but requires the ETT values on every link to be periodically estimated.

D. Routing Protocol

Suitable reactive or proactive routing strategies can be devised to implement the proposed routing heuristics. We now explain one possible implementation based on DSR, a reactive source-routed protocol. The source node broadcasts a route request (RREQ) packet. Any non-destination node that receives the route request packet (for the first time), rebroadcasts the packet after adding the appropriate costs (based on the heuristic being used) for the link over which RREQ was received, to the packet. The destination node sends a route reply (RREP) to the source node for every RREQ that it receives. The RREP contains all the cost information aggregated in the RREQ, and can be used by the source node to select the least cost route.

As we noted earlier, broadcast is more expensive with multiple channels because a copy of the packet has to be separately sent on each channel. The total broadcast cost can be reduced by using a two-phase route discovery process. In the first phase, each node forwards the RREQ packet only on the channel with the least cost. If a RREP is not received within a timeout interval, a second phase that involves a full route discovery (similar to the single-phase mechanism described above) is invoked. The two-phase route discovery process reduces the total broadcast cost when the first phase discovers at least one route. However, the discovered routes may not be optimal, as locally optimizing costs during the discovery process may not lead to a globally minimum cost route.

VI. DISCUSSION

In this section, we discuss other issues that may arise in multi-channel, multi-interface networks.

A. Impact of mobility

In the previous sections, we have not explicitly studied the impact of mobility. The main impact of mobility is that the neighbor set frequently changes. Protocols may have to be designed to be resilient to changes to neighbor set. For example, some of the interface assignment and routing strategies that we proposed are resilient to changes to the neighbor set. Consider the "Hello" packet mechanism used to periodically discover the fixed channels of neighbors, and for balancing the fixed channel assignment in the neighborhood. The "Hello" packet mechanism automatically handles changing neighbor sets. High levels of mobility may require more frequent "Hello" packet exchange increasing the overhead, but the overheads will still consume a very small fraction of the available channel bandwidth. Similarly, other protocols may be designed that are suitable for use even in mobile topologies. Another impact of mobility is the possibility of higher channel fading, leading to link breakages. It may be possible to exploit the resilience multiple channels offer against channel fading by developing a suitable interface assignment protocol.

B. Topology Control

The performance of wireless MAC protocols such as IEEE 802.11 significantly degrades when the number of contending transmissions increases. Many topology control strategies have been proposed for dense networks to reduce the number of contending transmissions, for example by using transmission power control. The use of multiple channels offers a similar benefit by distributing nodes across channels, thereby reducing the average number of contending nodes in a neighborhood by a factor of N, where N is the number of channels available. A carefully designed interface assignment strategy, along with a suitable routing algorithm, can dynamically adapt to the density of nodes in a neighborhood. If the node density is low, connectivity is maintained by using frequent interface switching. If the node density is high, sufficient connectivity is obtained without frequent interface switching, and the routing algorithm will mostly use routes that incur little switching cost. An open issue is to integrate protocols for multiple channels with transmission power control approaches for topology control.

C. Other issues with multiple channels

In this paper we have argued that multiple interfaces are useful for exploiting multiple channels. One open question is the number of interfaces that are needed for achieving maximum capacity improvement. Note that if N channels are available, then for the simultaneous use of the N channels, we need at least $2 * N$ interfaces (a pair of interfaces are required for communication on each channel). Thus, in any neighborhood (neighborhood is informally defined as the a region where any two communications on the same channel interfere), the total number of interfaces available among all nodes in the neighborhood has to be at least $2 * N$. If the total number of interfaces is less than that, then the *lack of sufficient number of interfaces* will be a bottleneck to performance. On the other hand, if the total number of interfaces is significantly larger than $2 * N$, then the *contention on the channels* will be a bottleneck to performance. Thus, selecting the number of interfaces M each node should have depends on the network density, topology, and the desired cost or performance.

Multiple channels may be used to derive other benefits. For example, we have proposed to use a single-path routing algorithm. In single channel networks, multi-path routing algorithms are often not effective as the chosen paths have to be *interference-disjoint* (i.e., the paths should not interfere on the wireless channel), and it is often difficult to find such paths. On the other hand, if multiple channels are available, then it is sufficient for the paths to be *node-disjoint*, as it may be possible to select routes that use different channels. When the node density is high, the number of node-disjoint paths may be large, while the number of interference-disjoint paths is still small. Hence, multiple channels may simplify the use of multi-path routing algorithms.

VII. CONCLUSION

In this paper, we have argued that capacity improvements with multi-channel networks can be exploited even when the number of available interfaces is smaller than the number of available channels. We have presented an interface assignment strategy that allows nodes to communicate with each other in a multi-channel environment without requiring specialized coordination algorithms. We have identified the need for specialized routing protocols for multi-interface networks, and have proposed routing heuristics that includes the impact of switching delay.

On-going work is studying alternate metrics and routing strategies for multi-channel, multi-interface networks. Detailed performance evaluation is also part of on-going work.

REFERENCES

[1] *IEEE Standard for Wireless LAN-Medium Access Control and Physical Layer Specification, P802.11*, 1999.

[2] Paramvir Bahl, Atul Adya, Jitendra Padhye, and Alec Wolman, "Reconsidering Wireless Systems with Multiple Radios," *ACM Computing Communication Review*, July 2004.

[3] A. Nasipuri, J. Zhuang, and S.R. Das, "A Multichannel CSMA MAC Protocol for Multihop Wireless Networks," in *WCNC*, September 1999.

[4] A. Nasipuri and S.R. Das, "Multichannel CSMA with Signal Power-based Channel Selection for Multihop Wireless Networks," in *VTC*, September 2000.

[5] N. Jain, S. Das, and A. Nasipuri, "A Multichannel CSMA MAC Protocol with Receiver-Based Channel Selection for Multihop Wireless Networks," in *IEEE International Conference on Computer Communications and Networks (IC3N)*, October 2001.

[6] Shih-Lin Wu, Chih-Yu Lin, Yu-Chee Tseng, and Jang-Ping Sheu, "A New Multi-Channel MAC Protocol with On-Demand Channel Assignment for Multi-Hop Mobile Ad Hoc Networks," in *International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN)*, 2000.

[7] Wing-Chung Hung, K.L.Eddie Law, and A. Leon-Garcia, "A Dynamic Multi-Channel MAC for Ad Hoc LAN," in *21st Biennial Symposium on Communications*, Kingston, Canada, June 2002, pp. 31–35.

[8] Jungmin So and Nitin H. Vaidya, "Multi-channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals using a Single Transceiver," in *Mobihoc*, 2004.

[9] Atul Adya, Paramvir Bahl, Jitendra Padhye, Alec Wolman, and Lidong Zhou, "A Multi-Radio Unification Protocol for IEEE 802.11 Wireless Networks," in *IEEE International Conference on Broadband Networks (Broadnets)*, 2004.

[10] Paramvir Bahl, Ranveer Chandra, and John Dunagan, "SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad-Hoc Wireless Networks," in *ACM Mobicom*, 2004.

[11] Richard Draves, Jitendra Padhye, and Brian Zill, "Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks," in *ACM Mobicom*, 2004.

[12] N. Shacham and P. King., "Architectures and Performance of Multi-channel Multihop Packet Radio Networks," *IEEE Journal on Selected Area in Communications*, vol. 5, no. 6, pp. 1013– 1025, July 1987.

[13] Jungmin So and Nitin H. Vaidya, "A Routing Protocol for Utilizing Multiple Channels in Multi-Hop Wireless Networks with a Single Transceiver," Tech. Rep., University of Illinois at Urbana-Champaign, October 2004.

[14] Ashish Raniwala, Kartik Gopalan, and Tzi-cker Chiueh, "Centralized Channel Assignment and Routing Algorithms for Multi-Channel Wireless Mesh Networks," *Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, April 2004.

[15] Ashish Raniwala and Tzi-cker Chiueh, "Architecting a High-Capacity Last-Mile Wireless Mesh Network," in *ACM Mobiciom (Poster)*, 2004.

[16] Marco Ajmone Marsan and Fabio Neri, "A Simulation Study of Delay in Multichannel CSMA/CD Protocols," *IEEE Transactions on Communications*, vol. 39, no. 11, pp. 1590–1603, November 1991.

[17] Ranveer Chandra, Paramvir Bahl, and Pradeep Bahl, "MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card," in *IEEE Infocom*, Hong Kong, March 2004.

[18] "Maxim 2.4 GHz 802.11b Zero-IF Transceivers," http://pdfserv.maxim-ic.com/en/ds/MAX2820-MAX2821.pdf.

[19] David B. Johnson, David A. Maltz, and Yih-Chun Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *Ietf Manet Working Group (Draft 10)*, 2004.

[20] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," in *Ietf RFC 3561*, July 2003.