Robust Floor Acquisition in the Presence of Multiple Fading Channels

Min Cao, Vivek Raghunathan, and P. R. Kumar

Abstract*— In this paper, we investigate the impact of multi-path fading on the floor acquisition mechanism of IEEE 802.11, and exploit multi-channel diversity to combat its adverse effects. We first provide experimental data to show that multi-path fading effects observed at the MAC layer appear at timescales that can interact negatively with the RTS-CTS-DATA-ACK handshake. These multi-path fading effects often result in control packets being destroyed, and cause the handshaking mechanism to malfunction. We identify two problems that result from this negative interaction, viz., the *fading exposed and fading hidden terminal problems*. Testbed measurements indicate that these multipath fading effects are statistically uncorrelated across different frequency channels. The presence of such multiple independent channels provides an additional degree of freedom to mitigate the negative impact of multipath fading. We exploit this multi channel diversity to design SIX, a *state information exchange* protocol that distributes MAC state information among neighbors. This information can be used to undo mistakes made in setting the IEEE 802.11 network allocation vector (NAV), a procedure we term *NAV overriding*. Furthermore, the transmitter can use this information to do *receiver selection* to the neighbor with the best link condition. These mechanisms combine to provide a robust floor acquisition mechanism that exploits *MAC layer diversity* in lossy environments. We evaluate the effectiveness of the SIX scheme with ns-2 simulations, and show that it can successfully harness multi-channel diversity to achieve reliable floor acquisition and improve performance as compared to the baseline IEEE 802.11 MAC.

Keywords: multi-path fading, floor acquisition, multi-channel diversity, IEEE 802.11, MAC

I. Introduction

There are two fundamental challenges in the design of wireless networks: *fading* and *interference*. Fading is the time variation of the wireless channel due to large-scale effects of path loss and shadowing, and small-scale effects of multi-path scattering. Fading is traditionally mitigated in the physical layer by coding and modulation techniques that exploit time, frequency, code and space diversities. The goal is to provide a reliable point-to-point communication link to the MAC layer. Interference arises from the shared nature of wireless channel where a point-to-point transmission may be corrupted by other transmissions in the vicinity of the receiver. Conventionally, interference is handled by the medium access control (MAC) layer. In IEEE 802.11, a four-way RTS-CTS-DATA-ACK handshake is used to acquire the channel, where the RTS silences potential competing nodes in the sender's neighborhood, and CTS silences potential senders in the vicinity of the receiver. This relies on the assumption that the physical layer can perfectly hide the fading effects from the MAC so that the probability of control packets (RTS/CTS) getting lost is very small and thus the impact of their loss on channel acquisition is negligible.

In this paper, we begin by summarizing an earlier experimental study that shows that this is not the case, and that fading can, in fact, be seen at the MAC layer. Depending on the environment, this fading can interact with the timescales of IEEE 802.11. This coupling between fading effects and IEEE 802.11 handshaking can result in two problems [1]:

1) The *fading exposed terminal problem* arises when fading destroys the RTS to the intended receiver, but not to other neighboring nodes. This causes nodes to unnecessarily defer from transmitting even though there is no ongoing transmission on the channel.
2) The *fading hidden terminal problem* occurs when nodes in the vicinity of the transmitter/receiver do not overhear the RTS/CTS and proceed to transmit on the channel, thereby destroying the ongoing transmission.

In this work, we resort to *multi-channel diversity* to combat this negative impact of fading on the MAC layer handshake. In many wireless systems, there are multiple available non-overlapping channels. For example, IEEE 802.11b/g has three such channels in the 2.4 GHz band and IEEE 802.11a has thirteen in the 5 GHz band. Experimental measurements indicate that the multipath fading effects experienced in different frequency channels are statistically uncorrelated. Under the assumption that it is possible to transmit/receive in different frequency bands simultaneously, this multi-channel diversity can be harnessed to provide more reliable handshaking. "Side information" can be passed over other channels to neighboring nodes when one of the channels is in a "deep fade". We use this idea to design a link layer *State Information eXchange* (SIX) protocol, where the MAC state information of each node is propagated to its neighbors through piggybacking. Nodes can exploit this information to better distinguish interference from fading and correctly set the network allocation vector (NAV), thereby increasing the robustness of the MAC handshake. The MAC state information of neighboring nodes can also be used by a transmitter to opportunistically select the receiver with the best link condition at any time among the multiple intended receivers. This *receiver selection* approach can be expected to boost performance in multipath fading environments, since it harnesses multi-receiver fading diversity.

This material is based upon work partially supported by DARPA/AFOSR under Contract No. F49620-02-1-0325, AFOSR under Contract No. F49620-02-1-0217, USARO under Contract No. DAAD19-01010-465, NSF under Contract Nos. NSF ANI 02-21357 and CCR-0325716, and DARPA under Contact No. N00014-0-1-1-0576, and Vodafone Graduate Fellowship.

There has been a lot of related work in the area of cross layer PHY/MAC design of wireless networks [2], [3] and the design of MAC protocols that use multiple channels [1], [4], [5]. We first identified the negative coupling between multipath fading and IEEE 802.11 handshaking as a potential problem in the context of the design and evaluation of the fading-aware DB-MCMAC multi-channel MAC protocol [1]. The interaction between multipath fading and IEEE 802.11 handshaking does not seem to have been addressed elsewhere in the literature. To the best of our knowledge, our work is the first attempt to use multi-channel diversity to improve the robustness of IEEE 802.11 floor acquisition in multipath fading environments.

The remainder of this paper is organized as follows: In Section II, we discuss the impact of fading on the MAC layer handshake, present experimental evidence of fading effects seen at the MAC layer, and motivate the use of multi-channel diversity to mitigate their impact. In Section III, we propose SIX, a link layer *State Information eXchange* protocol that provides robust channel acquisition by exploiting the MAC layer diversities. The *ns-2* simulation results are discussed in IV and we conclude in Section V.

II. Motivation

A. Fading effects at the MAC layer

We start off by using experimental data to demonstrate that multipath fading effects can be observed at the MAC layer. Since the physical layer uses coding to mitigate the effects of fading, the MAC layer sees the wireless channel between any two nodes as being in one of two states: in the "good" state, all packets get through without loss, and in the "bad" state, all packets are dropped. A popular model for this kind of fading behavior is a two state Gilbert Eliot Markov fading process with the key parameters being the time spent in the "good" and "bad" states respectively.

As part of a more exhaustive study into the interaction of multipath fading and the IEEE 802.11 MAC [1], we carried out detailed experiments to obtain typical values of these parameters in IEEE 802.11 environments. IEEE 802.11b takes (approximately) 12 ms to deliver a 1500 byte packet at a data rate of 1 Mbps. In this experimental study, we observed both slow and fast fading relative to this IEEE 802.11 timescale. For example, in the experimental run shown in Figure 1, the fading timescale is concentrated in the 10 to 100 ms range. In other runs, we observed fading in the 1 to 10 ms timescale too. When the fading timescale is of the order of $1 - 10$ ms ("fast fading"), the multipath fading effects interact with the IEEE 802.11 MAC layer handshaking on a very fine granularity. This problem becomes less pronounced when the fading timescales increases to 100ms and more, but still interacts with the MAC layer handshaking in a negative way, as described in [1].

We now present experimental data to indicate that the multipath fading effects are roughly uncorrelated across different frequency channels. Our experimental setup consists of two collocated IEEE 802.11 transmitters sending one byte ICMP ECHO REQUEST broadcast probes at 1 mW to two collocated receivers on channels 1 and 10 respectively. We use 3M 1181 EMI copper shielding tape to reduce the effective transmit

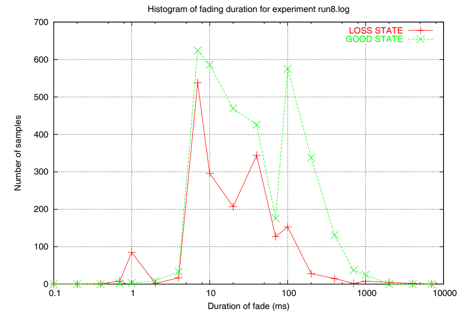

Fig. 1. Histogram of MAC fading timescale: slow fading.

range [6], and turn off ICMP ECHO RESPONSE at the receivers to kill reverse direction traffic. We log the received probes using tcpdump, taking care to synchronize the receivers using NTP. Let $X(t)$ and $Y(t)$ be the respective channel states for each transmitter-receiver pair as a function of time, inferred from the sequence number information on received probes. Then, the cross correlation, defined as $\rho(X,Y) = \frac{cov(X,Y)}{\sigma_X \sigma_Y}$, is a good measure of the statistical relationship between fading on the two channels. We estimated this cross correlation for multiple runs, with each run from three to five minutes. The mean of the absolute value of this cross correlation was 0.05548, and all experimental runs had cross correlation in $[-0.004, 0.2718]$. The low mean value of $|\rho(X,Y)|$ indicates that multipath fading has very little statistical correlation across different channels, and that different frequency channels approximately fade independently. This diversity across channels provides an additional degree of freedom to leverage in mitigating the interaction of fading with IEEE 802.11, which we exploit in designing the SIX protocol.

B. Negative impact of fading on channel acquisition

In IEEE 802.11, channel acquisition is achieved through the four-way RTS-CTS-DATA-ACK handshake. However, when the wireless channel is in "deep fade", RTS/CTS control packets can be destroyed with high probability. In environments with multiple receivers, the multipath fading effects are independent for different (transmitter, receiver) pairs; and as a result, it is possible that some of the neighbor nodes get the RTS/CTS, while others do not. This results in the malfunction of the RTS-CTS channel acquisition mechanism [1].

Consider the scenario shown in Fig.2. At time t_0, node C sends RTS$_{C \to D}$ to node D. Suppose that at t_0, link l_{CD} is in deep fade while link l_{CB} is in good state. Then, node C gets no response from D, while node B receives RTS$_{C \to D}$ and sets its NAV accordingly. Assume that node C exceeds its retry limit and gives up its attempted transmission to D, and at time t_1 node A sends RTS$_{A \to B}$. Node B cannot reply with CTS since it is silenced by node C, although there is actually no conflict since node C has given up its attempt. We call this the *fading exposed terminal problem*.

Correspondingly, fading can also cause a *fading hidden terminal problem*, as seen in Fig.3. In this scenario, at time t_0, node A sends RTS$_{A \to B}$ to node B, which replies with CTS$_{B \to A}$ in time t_1. Suppose that node C fails to decode CTS$_{B \to A}$ due to multipath fading and thus, does not update

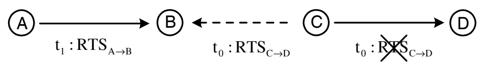

Fig. 2. Fading exposed terminal scenario.

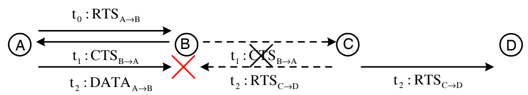

Fig. 3. Fading hidden terminal scenario.

its NAV to defer for the DATA transmission from A to B. Suppose at time t_2, node C sends $\text{RTS}_{C\to D}$ to node D, it will collide with $\text{DATA}_{A\to B}$ at receiver B. This problem could also occur when RTS fails to silence its neighboring nodes. We call this the *fading hidden terminal problem*. Note that the RTS/CTS could also be lost due to interference instead of fading, resulting in the *interference exposed/hidden terminal problems* respectively.

These problems are intrinsic to handshaking MAC protocols. It is typically assumed in the literature that the probability of control packets getting lost is very low so that these problems have a negligible impact on system performance. However, in lossy multipath fading environments, they become more significant and need to be mitigated. The key observation we make is that if each node has "side information" of the MAC state of its neighbor nodes, then the above problems can be solved. In the fading exposed terminal scenario seen in Fig.2, node B can reply to A's RTS with a CTS if it knows that node C has given up its RTS attempt. Similarly, in the fading hidden terminal scenario shown in Fig.3, node C will set its NAV properly if it knows that node B has sent a CTS and is receiving a DATA packet. Such side information could be obtained by explicitly exchanging additional control messages in the same channel. However, this would make the handshaking more complicated, and these control messages are still likely to be destroyed when the channel experiences deep fading. Instead, our approach, described in the next section, is to take advantage of multiple independent fading channels whenever they are available and piggyback this side information over other channels. By doing so, we can provide a robust channel acquisition scheme in lossy multi-channel fading environments.

C. MAC layer diversity

Fading effects are traditionally mitigated in the physical layer through the use of time/frequency diversity. The presence of multiple receivers and channels provides another form of diversity that can be exploited in the MAC layer to mitigate fading and interference [1]. For example, consider the scenario shown in Fig.4, where there are two available channels and node A has packets for both neighbors B and C. Let l_{mn}^{Chi} denote the link on channel i from transmitter m to receiver n. The link conditions on each l_{mn}^{Chi} could be varying, either due to multipath fading effects or interference at the intended receiver. When a packet cannot get through link l_{mn}^{Chi}, the link is said to be in the "bad" state, otherwise it is in the "good" state. Since the link conditions of different links vary independently in practice, it is possible that although the link l_{AC}^{Chi} from A to C is "bad", the link l_{AB}^{Chi} from A to B is "good"

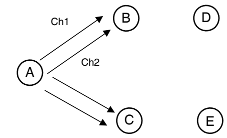

Fig. 4. A MAC diversity scenario.

, or vice versa. This is termed as *multi-receiver diversity* and can be opportunistically exploited by a MAC that selects the receiver with the best link condition at any instant. Similarly, the presence of multiple independent fading channels between a transmitter and receiver provides *multi-channel diversity* and allows the transmitter to opportunistically choose the channel with the best link condition at every instant.

In order to efficiently harness these diversities, the MAC layer should be able to track the channel fading and interference conditions on a per neighbor and per channel basis. The state exchange protocol described in the next section allows a node to obtain "side information" of its neighbor nodes' MAC states. The node can thus track link interference conditions, and opportunistically harness MAC diversity. For example, in the scenario in Fig.4, node A should transmit packets to B when it is idle and node C is silenced by E, and transmit to C when B is silenced by D. In IEEE 802.11, a RTS packet to a node that has been silenced by other nodes leads to costly exponential backoff, which can be avoided by exploiting "side information" provided by our scheme. In our *receiver selection* scheme described below, the wireless device driver at each node maintains a per neighbor queue and uses the channel conditions to intelligently send packets down to the MAC.

III. Proposed Scheme

In this section, we propose SIX, a link layer *State Information eXchange* protocol to provide robust channel acquisition and exploit MAC layer diversity. The architecture of SIX is shown in Fig.5. In the MAC layer, there is a programmable IEEE 802.11 DCF interface on each channel. In the IFQ, per neighbor FIFO queues are implemented. We do not change the RTS-CTS-DATA-ACK handshaking mechanism. However, we assume that it is possible to change some of the internal operations of the 802.11 MAC (e.g., the NAV setting mechanism); and that we can piggyback some bits in the MAC control packets (RTS/CTS/ACK), or create a broadcast packet to exchange MAC state information among neighbors. A *Neighbor Information* module is implemented at the link layer to maintain the MAC state information for each neighbor, and is updated upon receiving a piggybacked control packet or broadcast packet with MAC state information. We illustrate how this information can be used to provide a more reliable and efficient link layer through the use of two simple techniques. Firstly, the MAC layer at each node updates the NAV setting on each channel according to the updated MAC states of neighbor nodes, a procedure we term *NAV overriding*. Second, the IFQ intelligently selects the intended receiver based on their corresponding MAC states before passing a packet to the MAC, a mechanism we call *receiver selection*.

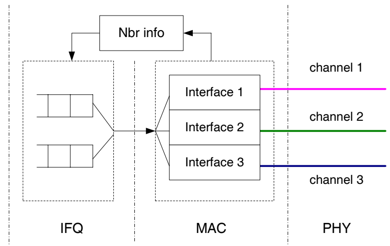

Fig. 5. Architecture of the SIX scheme.

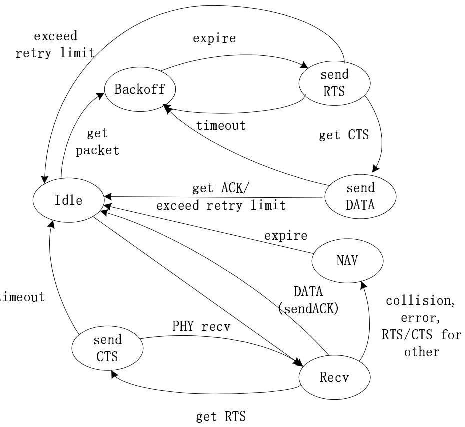

Fig. 6. Simplified state machine of the IEEE 802.11 MAC.

A. MAC state information

We must first identify the MAC state information that is to be exchanged among neighboring nodes. It is known that the IEEE 802.11 MAC interface's operation can be described as a finite state machine. A simplified state machine is shown in Fig.6, where we neglect the inter-frame spacing and regard the ACK as part of the DATA transmit/receive process. Each interface's MAC can be in one of the following states: *Rts, Cts, Send, Recv, Backoff, Nav, Idle*. We describe the MAC state information of each interface i as a tuple $\langle MacState_i, ExpireTime_i\rangle$, where $MacState_i$ is the current MAC state of the interface and $ExpireTime_i$ is the time duration that the interface will stay in state $MacState_i$ (except for RTS/CTS as described below). Let τ_{Sifs} denote the SIFS spacing time, τ_{Difs} denote the DIFS spacing time, and τ_{Rts}, τ_{Cts}, τ_{Data}, τ_{Ack} denote the length of RTS, CTS, DATA, ACK packets in IEEE 802.11 respectively and let the backoff time be denoted by τ_{BO}. When the interface i is ready to send a RTS, $ExpireTime_i$ is set to $\tau_{Rts} + \tau_{Cts} + \tau_{Data} + \tau_{Ack} + 3\tau_{Sifs}$, which is the duration for which the interface is expected to acquire the channel if the RTS is successful. The purpose is to distribute this information to neighbors through other channels, in case the RTS packet is lost in this channel. If the RTS gets through and the interface is ready to send the DATA, $ExpireTime_i$ is set to be $\tau_{Data}+\tau_{Ack}+\tau_{Sifs}$. If the RTS fails and the MAC enters into *Backoff* state, $ExpireTime_i$ is set to the expiration time of the backoff timer. When the retry limit is met and the data packet is discarded, the state is set to *Idle* and $ExpireTime_i$ is set to *Null*. Similarly, when CTS is to be sent, $ExpireTime_i$ is set to be $\tau_{Cts} + \tau_{Data} + \tau_{Ack} + 2\tau_{Sifs}$, and so on. When the NAV is set, $ExpireTime_i$ is set to the time at which the NAV will expire.

B. Message distribution

Since the fading effects on different channels are statistically uncorrelated in practice, we can explore this additional degree of freedom to pass the MAC state information of one channel to neighbor nodes over other channels. This will provide the side information for neighbor nodes to combat RTS/CTS loss and to explore MAC layer diversities. We employ two mechanisms to handle message passing: *piggyback* and *broadcast*. When we send a control packet (RTS/CTS/ACK) over one channel, we piggyback the MAC state information of every other channel in the control packet. The piggyback information for channel i is $\langle MacState_i, Duration_i\rangle$, where $Duration_i = \max\{0, ExpireTime_i - t_{now} - \tau_{Ctrl}\}$, τ_{Ctrl} is the length of the control packet and t_{now} denotes the current clock time. Note that $MacState_i$ takes a value on 8 states and can be represented with 3 bits. We use 13 bits to represent $Duration_i$ in terms of number of IEEE 802.11 slots (1 slot = 20 μs). This provides sufficient accuracy for the SIX protocol operation, where $Duration_i$ is typically on the order of ms. Thus, with N channels, the piggybacking overhead is $2(N-1)$ bytes. This overhead is acceptable, specially for small N, as the control packet length is typically 40 bytes in IEEE 802.11.

The piggybacking mechanism can efficiently distribute MAC state only if the traffic load is high enough that every interface on the node is busy competing for channel access by exchanging RTS/CTS/ACK frames. On the other hand, when the network is not loaded, i.e., many interfaces are often in the *Idle* state, the piggyback mechanism is not sufficient. In such an environment, we create a short broadcast packet containing MAC state information to be sent over the idle channels. Since the broadcast packet is very short, it will typically not cause a collision. We only create the broadcast packet if it is necessary to notify its neighbors of a MAC state change: 1) RTS or CTS is being sent on a channel, which corresponds to a transmit/receive attempt; 2) RTS/DATA retry limit is met so that the transmit attempt is aborted, or CTS timeout so that the receive attempt is aborted. In this case, we need to cancel the NAV setting. In order to keep the control overhead low, we broadcast the created MAC state packet with probability $P_{broadcast}$, where $P_{broadcast}$ is a parameter.

C. Neighbor information update

In the *Neighbor Information* module, MAC state information $\langle MacState_{i,k}, ExpireTime_{i,k}\rangle$ is maintained on a per neighbor, per channel basis, where the index (i, k) denotes channel i of neighbor node k. By default, $MacState_{i,k}$ is set to *Unknown*, and $ExpireTime_{i,k}$ to $Null$ for all i and k. When a node receives the MAC state information from a piggybacked control packet or broadcast packet from neighbor k, it updates its information in the Neighbor Information module accordingly, i.e., for each channel i, it sets $MacState_{i,k} = MacState_i$ and $ExpireTime_{i,k} = t_{now} + Duration_i$. In addition, we also monitor for the expiry time of the MAC states so that if $t_{now} = ExpireTime_{i,k}$, $MacState_{i,k}$ is set to *Unknown* and $ExpireTime_{i,k}$ to *Null*.

D. NAV overriding

When a node receives the MAC state information from a piggybacked control packet or broadcast packet from neighbor k, it updates the NAV setting of each of its interfaces accordingly. We define the compound state $Busy = \{Rts \vee Cts \vee Send \vee Recv\}$, where $\vee$ denotes the OR operation. Let t^i_{nav} be the current NAV expiry time of interface i. If in the received MAC state information, $MacState_i = Busy$ and $t_{now} + Duration_i > t^i_{nav}$, then the node updates its NAV expiration time as $t^i_{nav} = t_{now} + Duration_i$. In case the RTS or CTS packet over channel i from neighbor k was lost, this NAV update procedure will correct the NAV setting and help mitigate the fading hidden terminal problem discussed in Section II-B. On the other hand, if $MacState_i = Nav \vee Idle \vee$ *Backoff* and the current t^i_{nav} had been set by neighbor k, we have a fading/interference exposed terminal scenario. In this case, we should cancel the NAV set by neighbor k and recover the previous NAV setting. This requires the MAC to record the identity of the nodes that set the previous and current t^i_{nav}, as well as the actual value of the previous t^i_{nav}. By undoing the NAV setting in this manner, the fading/interference exposed terminal problem is solved once we obtain the MAC state information.

E. Receiver selection

We have noted earlier that MAC diversities can be exploited to enhance performance by selecting the receiver with the best link conditions for each channel. With the SIX protocol, we can opportunistically utilize neighbor MAC state information in the Neighbor Information module to implement intelligent *receiver selection*. In this mechanism, we implement per receiver queues in the IFQ on a per-neighbor basis. When one of the channels, say channel i, becomes available, the IFQ should select the neighbor with the link condition that indicates that the packet is most likely to get through. We check the Neighbor Information module to find a neighbor k with $MacState(i, k) = \{Idle \vee Backoff\}$ and non-empty receiver queue. If there is more than one such candidate neighbor, we choose the one with the largest preference index $p_{i,k}$, where $p_{i,k}$ is an estimate of the link conditions of l^{Chi}_k based on the past history of transmissions. While there are many possible approaches to set $p_{i,k}$, we simply set it to $p_{i,k} = 1/\nu_{i,k}$, where $\nu_{i,k}$ is the time that interface i spent on transmitting the last packet to neighbor k (if the last packet was dropped, then $\nu_{i,k}$ is the time difference between when the interface i got the packet and when it actually dropped it). If none of the neighbors are in *Idle* or *Backoff* states, we choose the neighbor with the largest $p_{i,k}$ whose $MacState_{i,k} = Unknown$. Finally, if all the neighbors are either in the *Busy* or *Nav* states, we infer that none of them will respond to a RTS and thus, the IFQ should wait till t_{hold}, where t_{hold} is determined as follows: if there exists a neighbor in the $Busy$ state, then $t_{hold} = \max_k\{ExpireTime_{i,k}|MacState_{i,k} = Busy\}$, else $t_{hold} = \min_k\{ExpireTime_{i,k}|MacState_{i,k} = Nav\}$.

IV. Performance Evaluation

We have implemented the SIX protocol in ns-2 and are carrying out an extensive simulation analysis. In this paper, we present a few illustrative plots.

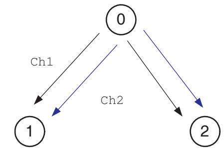

Fig. 7. Scenario 1: 1 sender, 2 receivers, 2 channels.

A. Simulation methodology

We have built a new radio propagation model in ns-2 to simulate the effect of small-scale fading. The wireless medium between any (node, neighbor, channel) 3-tuple is independent and assumed to be in one of two states. In the "good" state, the large scale two ray ground propagation model decides the received signal strength. In the "bad" state, the received signal strength is zero and all packets are dropped at the receiver. The transition between the two states is modelled as a two-state Markov fading process.

We compare SIX against a baseline scheme where each interface is tuned on a different channel and runs an independent instance of IEEE 802.11 DCF. Packets for all receivers are queued in a global FIFO queue. When an interface finishes a transmission, a new packet is dequeued from the global queue and transferred to the interface. Transport layer goodput is the primary metric we use to characterize performance. In all the simulations, we use a transmit range of 250m. Carrier sensing range is set to the transmission range. The IFQ length is set to 50 packets. The data rate is set to 1 Mbps. We use CBR sources with a size of 1500 bytes on top of UDP. Simulations are run for 100 seconds.

B. Collocated topology

First, we test a simple collocated topology with one sender, 2 receivers and 2 channels, as shown in Fig.7. The two state Markov model is used with fading timescales varying from 1ms to 1s. Each link fades independently. The system throughput of SIX and the baseline scheme are shown in Fig.8. It can be seen from the figure that SIX achieves improvements over the baseline ranging from 7% at a fading timescale of 1 ms to 81% at a fading timescale of 1 s. SIX obtains these improvements because it can reduce the occurrence of the fading exposed terminal problem in this case and exploits multi-receiver fading diversity. The performance improvement diminishes in fast fading because the state information exchange is not able to track the channel state variations in real-time.

C. Line topology

We proceed to next evaluate the performance of SIX with a simple 3 channel multi-hop topology shown in Fig.9, where there are four nodes in a straight line, and two UDP flows: one from node 0 to 1 and the other from node 2 to 3. (As noted earlier, this is a canonical scenario for the fading hidden terminal problem.) We compare the total throughput achieved by SIX and the baseline for different fading timescales in Fig.10. We

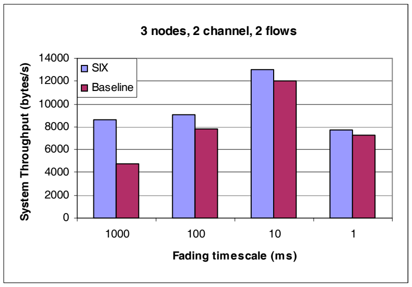

Fig. 8. Performance of SIX for scenario 1.

Fig. 9. Scenario 2: 4 nodes, line topology, 3 channels.

notice again that SIX obtains performance improvements over the baseline scheme in relatively slow fading environments (84% at 1 s timescale and 26% at 100 ms timescale), while the gains degrade for fast fading channels. This indicates that SIX can mitigate fading hidden terminal problems in slow fading environments where the state information exchange can track the channel state variations.

D. Random topology

Finally, we verify that SIX is able to efficiently harness MAC layer diversities to provide performance improvement in a wide range of general scenarios. We generate 40 random topologies and traffic flow patterns over a 1000m×1000m grid with 3 channels. Each scenario has 32 nodes and 20 UDP flows. Routing is done using DSDV, and fading is introduced at different timescales. The goodput is averaged over all scenarios and plotted in Figure 11. As seen from the figure, SIX achieves throughput improvements that increase monotonically from 4% at 10 ms fading timescale to 31% at 1 s timescale. This is consistent with the results in the collocated and line topologies.

V. Concluding Remarks

Traditionally, there has been a separation of functionality between the physical and MAC layers in wireless networks, with the physical layer handling fading and the MAC layer handling interference. Multipath fading effects are typically mitigated in the physical layer, while interference management is performed through four-way handshaking at the MAC layer. In this paper, we have demonstrated that the physical layer is not always able to perfectly hide fading from the MAC layer. In such lossy environments, fading effects can interact unfavorably with the RTS-CTS-DATA-ACK handshake, and cause the floor acquisition mechanism to malfunction, causing increased occurrence of exposed and hidden terminal problems.

To mitigate both fading and interference at the MAC layer, we must intelligently exploit the diversity available when there are multiple independent fading channels. We proposed SIX, a *state information exchange* protocol where the MAC state information is exchanged between neighboring nodes through piggyback or broadcast mechanisms. This information is used to combat RTS/CTS loss, and achieve more reliable channel

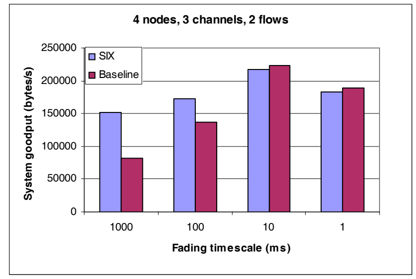

Fig. 10. Performance of SIX for scenario 2.

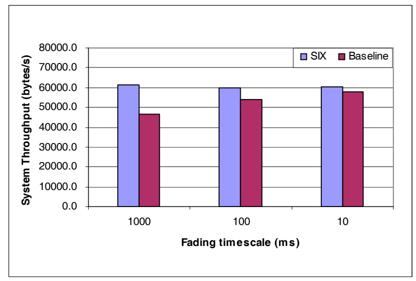

Fig. 11. Performance of SIX vs. baseline for general topologies

acquisition. The SIX framework also enables the exploitation of multi-receiver diversity when there are multiple intended receivers. We have evaluated the performance of SIX through ns-2 simulations. These simulations indicate that SIX is able to provide more robust channel acquisition as compared to IEEE 802.11, provided the fading is not too fast. We must emphasize that the SIX scheme is an exploratory attempt at resolving the negative interactions between fading and IEEE 802.11, and the design of efficient solutions to this problem is a big research challenge.

References

[1] M. Cao, V. Raghunathan, and P. R. Kumar, “Cross-layer exploitation of MAC layer diversity in wireless networks,” *submitted to ACM MOBIHOC*, 2006.

[2] G. Dimic, N. Sidiropoulos, and R. Zhang, “Medium access control - physical cross-layer design,” *IEEE Signal Processing Magazine*, 2004.

[3] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, “Opportunistic media access for multirate ad hoc networks,” in *Proceedings of ACM MOBICOM*, 2002.

[4] N. Jain, S. Das, and A. Nasipuri, “A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks,” in *Proceedings of IEEE ICCCN*, 2001.

[5] S. Wu, Y. Tseng, C. Liu, and J. Sheu, “A multi-channel MAC protocol with power control for multi-hop mobile adhoc networks,” *Computer Journal*, vol. 45, no. 1, 2002.

[6] V. Kawadia and P. R. Kumar, “Experimental investigations into TCP performance over wireless multihop networks,” in *Proceedings of ACM E-WIND*, 2005.