Message in Message (MIM): A Case for Reordering Transmissions in Wireless Networks

Naveen Santhapuri
Univ. of South Carolina

Justin Manweiler
Duke University

Souvik Sen
Duke University

Romit Roy Choudhury
Duke University

Srihari Nelakuduti
Univ. of South Carolina

Kamesh Munagala
Duke University

ABSTRACT

Message in Message (MIM) is an exciting development at the physical layer of IEEE 802.11. Two transmissions that otherwise conflict with each other, may be made concurrent with MIM. However, the benefits from MIM are not immediate. Higher layer protocols need to be explicitly designed to enable its inherent concurrency. This paper investigates the opportunities and challenges with MIM, and demonstrates a link layer framework to harness its potential. We believe that our framework can accommodate emerging physical layer capabilities, such as successive interference cancellation (SIC).

1. INTRODUCTION

Physical layer research continues to develop new technologies to better cope with wireless interference. One exciting development in the recent past is *Message in Message* (MIM). Briefly, MIM allows a receiver to disengage from an ongoing signal reception, and engage onto a new, stronger signal. What could have been a collision at the receiver, may now result in a successful communication. To better understand MIM, we compare it with the traditional notion of *collision* and *physical layer capture*. We refer to Figure 1 to describe this contrast. *We assume throughout the paper that the signal of interest (SoI) is sufficiently stronger than the interference*[1].

Collision was widely interpreted as follows: An SoI, however strong, cannot be successfully received if the receiver is already engaged in receiving a different (interfering) signal. Most simulators adopt this approach, pronouncing both the frames corrupt [1,2]. Figure 1(c) and (d) illustrate these cases. **Physical Layer Capture** was later understood through the systematic work in [3,4]. Authors showed that capture allows a receiver to decode an SoI in the presence of interference, provided the start of both the frames are within a preamble time window. While valuable in principle, the gains from capture are limited because the 802.11 preamble persists for a short time window (20 μs in 802.11a/g). If the SoI arrived 20 μs or later, both frames will still be corrupt (Figure 1(d)).

Message in Message (MIM) is empowering because it enables a receiver to decode an SoI, even if the SoI arrives after the receiver has already locked on to the interference [5]. Moreover, if the SoI arrives earlier than the interference, MIM allows reception at a lower SINR in comparison to a non-MIM receiver. Figures 1 (a), (b) and (d) identify these benefits over capture.

Two main ideas underpin the feasibility of MIM:
(i) An MIM receiver, even while locked onto the interference, simultaneously searches for a new (stronger) preamble. *If a stronger preamble is detected, the receiver unlocks from the ongoing reception, and re-locks on to this new one.* Of course, re-locking requires a higher SINR.

(ii) MIM takes advantage of the characteristics of interference signals. A strong decodable interference is better than a weak non-decodable interference, because *the ability to decode the interference enables the ability to suppress it as well*. As a result, an MIM receiver that is locked onto the interference is better equipped to suppress it, and re-lock onto the later-arriving SoI. Exploiting the same idea, if the receiver has locked onto the SoI first, and a weaker interference arrives later, the receiver is able to better "tolerate this distraction". As a result, the SoI can be received at a lower SINR.

Link Layer Opportunity

Unless guided by link layer protocols, the intuitive benefits from MIM may not translate into throughput improvements. We argue this using the example in Figure 2. When using MIM receivers, observe that the two links can be made concurrent only if AP1→R1 starts before AP2→R2. Briefly, since R2 satisfies a higher SINR of $20dB$, it can afford to decode its SoI even in the presence of interference from AP1. However, since R1 satisfies a lower SINR, starting earlier helps in locking onto AP1's signal in the clear. Had the order of transmission been reversed, AP1→R1 transmission would experience a collision. As a generalization of this example, MIM-aware scheduling protocols need to *initiate weaker links first, and stronger links later*. In a larger network, choosing the appropriate set of links from within a collision domain,

---

[1] We also assume that the interference is strong enough that, in the absence of the SoI, it can be decoded by the receiver.

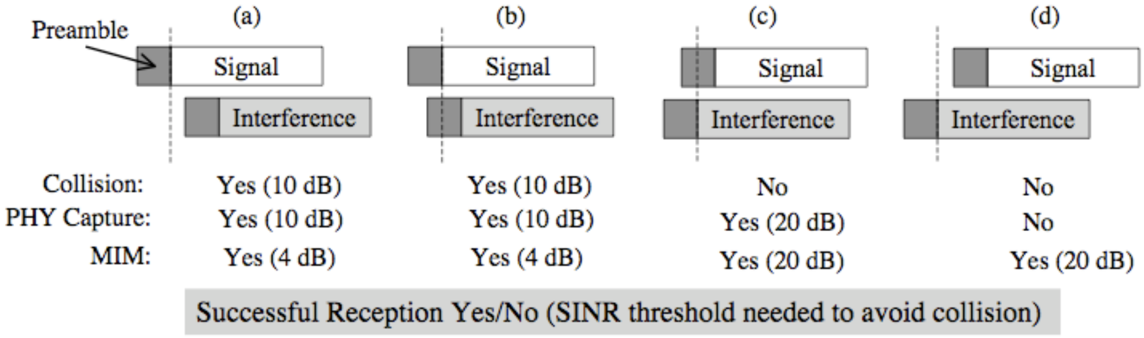

Figure 1: Evolving notion of successful reception. SINR values are approximate, and vary across hardware.

and determining the order of optimal transmission is a non-trivial research problem. IEEE 802.11 or other MAC protocols that are unaware of MIM [6] do not ensure such orderings, failing to fully exploit MIM-capable receivers. Perhaps more importantly, *graph coloring based scheduling approaches may also be inapplicable*. This is because graph coloring approaches assume symmetric conflicts between links. Link conflicts are *asymmetric* under MIM (i.e., depend on relative order), and may not be easily expressed through simple abstractions.

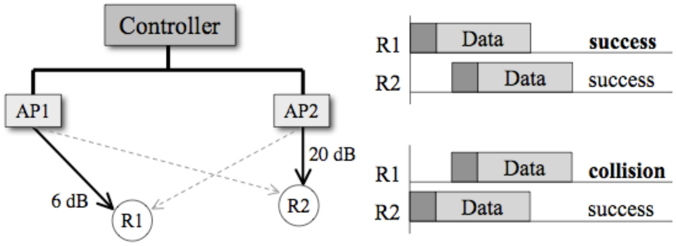

Figure 2: AP1→R1 must start before AP2→R2 to ensure concurrency. In the reverse order, R1 cannot lock onto AP1's signal because of AP2's interference.

In response to this rich research problem, this paper proposes an MIM-aware link layer solution that reorders transmissions to extract performance improvements. Our system is named *Shuffle* in view of its ability to shuffle the order of transmissions. Our main contributions are:

(1) Validation of MIM through experiments on a small testbed. We use MIM-enabled IEEE 802.11 compatible Atheros 5213 chipsets, running MadWiFi drivers on Soekris hardware. We show that order of transmission matters while decoding packets.

(2) Analysis of optimal performance improvements with MIM. We show that MIM-aware scheduling is NP-hard, derive upper bounds on throughput using integer programming in CPLEX, and design heuristics to attain these bounds.

(3) Design of an MIM-aware scheduling framework, *Shuffle*, for enterprise wireless LANs. Our approach to reordering transmissions offers consistent throughput improvements against both 802.11 and a centralized scheduling protocol. The subsequent sections expand on each of these contributions.

2. TESTBED MEASUREMENTS

We confirm the occurrence of MIM on a testbed of Soekris devices equipped with Atheros 5213 chipsets using the MADWiFi driver. Apart from corroborating the observations of [5] about MIM, we also show that due to MIM the order of transmission matters for successful delivery of packets. The experiment consists of two transmitters with a single receiver placed at various points in-between. This subjects the receiver to varying SINRs. To ensure continuous packet transmissions from the transmitters, we modify the MADWiFi driver to disable carrier sensing, backoff, and the inter-frame spacings (EIFS and SIFS). To time-stamp transmissions, a collocated receiver is placed at each transmitter. Using these time-stamps, we are able to merge multiple traces to determine which packets overlap in time and the relative order of overlap. We omit several implementation details, especially those related to achieving μs-granularity time synchronization among collocated receivers.

Figure 3 shows delivery ratio for different order of packet arrivals, at different positions of the receiver. For all these positions, the interference was strong, i.e., in the absence of the SoI, the interfering packets were received with high delivery ratio. Under these scenarios, observe that when the receiver is very close to the transmitter (positions 1, 2, and 3), it achieves a high delivery ratio independent of the order of reception. This is a result of achieving a large enough SINR, such that both SoI-first (SF) and SoI-last (SL) cases are successful. However, when the receiver moves away from the transmitter (positions 4 and 5), the SINR is only sufficient for the SF case, but not the SL case. Hence, only 4% of the late-arriving packets get received, as opposed to 68% of the early-arriving packets. This is a clear validation of MIM, and can be translated into throughput gains by deliberately regulating the start of packets.

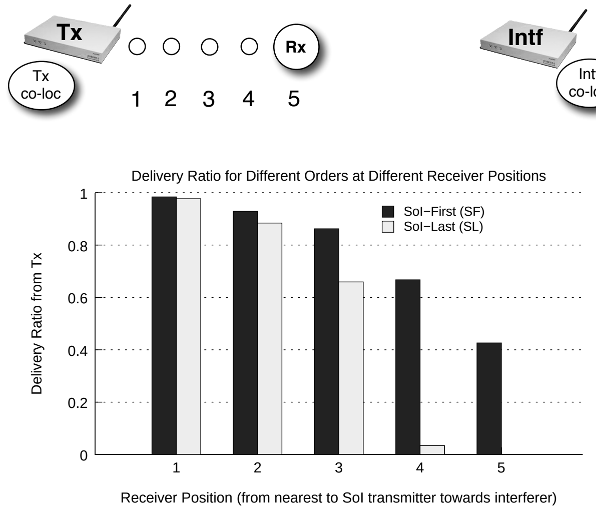

Figure 3: Testbed validates MIM. Rx receives from Tx (at 5 positions) in presence of interferer (Intf).

3. OPTIMALITY ANALYSIS

A testbed prototype validates the practicality of MIM, and indicates potential for gains in a large scale network. The natural question to ask is, *what is the maximum gain available from MIM?* Towards this, we perform an optimality analysis of MIM-capable networks. In the interest of space, we present only the main results.

Given a network, $G = (V, E)$, an MIM-enabled link scheduling algorithm needs to choose a suitable subset of links, $l \in E$, and specify their order of activation, O_l. The links and their order must be chosen such that the network concurrency (or network throughput) is maximized. We have shown that *Optimal MIM-aware scheduling is NP-hard*. The proof is derived through a reduction of the *Independent Set* problem, known to be NP-complete.

We developed an Integer Program (IP) to characterize the upper bounds on throughput for practical MIM-capable network topologies. The bounds are compared with an optimal MIM-incapable scheduling scheme (also NP complete). We omit the IP formulation, and only include the key results here. Figure 4 presents graphs generated using the CPLEX optimizer. Evident from the graphs, the ideal benefits from MIM can be high, especially for networks of realistic size.

4. SHUFFLE: AN MIM-AWARE LINK LAYER

We now propose Shuffle, a link layer solution that exploits MIM capabilities by carefully reordering transmissions. Shuffle targets enterprise WLAN (EWLAN) environments, such as universities, airports, and corporate campuses [7,8]. In EWLANs, multiple access points (APs) are connected to a central controller through a high speed wired backbone (Figures 2 and 6). The controller, besides acting as the gateway for Internet traffic, coordinates the operations of APs. The (thin) APs follow the controller's instructions for control and data packet communication.

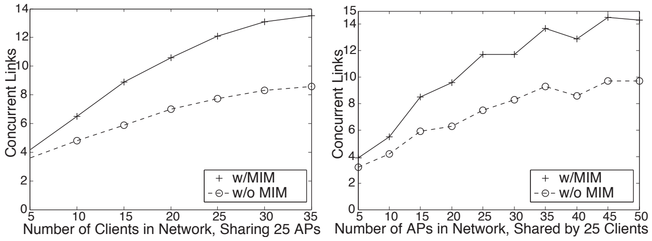

Figure 4: MIM performance bounds using CPLEX.

4.1 Protocol Architecture

Shuffle executes three main operations as follows: (1) Pair-wise interference relationships between links are characterized through a measurement-based procedure that we call *rehearsal*. (2) Utilizing the interference map, an MIM-aware scheduler (hosted at the EWLAN controller) computes a set of concurrent links, and their relative order of transmission. (3) A transmission manager coordinates the APs to execute the ordered schedule, and handles failures due to time-varying channel conditions. The rest of this section presents the details of these individual operations.

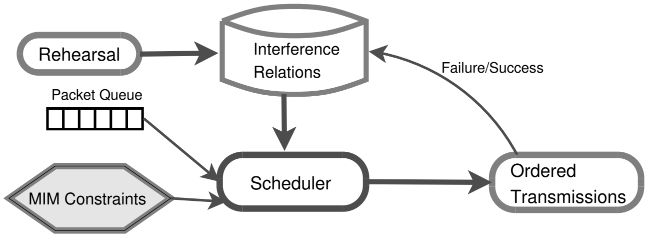

Figure 5: Architecture of Shuffle

4.1.1 *Rehearsal*

Scheduling algorithms require the knowledge of interference relationships between links. We propose a measurement-based approach to obtain this relationship. Specifically, the EWLAN controller coordinates the APs and clients to transmit probe packets in a systematic manner. Other clients and APs sniff the RSSI values for each transmission, and feed them back to the controller. The timing of transmissions are carefully planned to ensure clear measurements. Once all the link RSSI values are accumulated, the controller merges them into an interference map. For the merging operation, we assume that interference is linearly additive [9, 10]. Hence, once the rehearsal is over, the approximate SINR of any transmission, in the presence of other interfering transmissions, can be computed. Of course, these values change due to the time-varying nature of the wireless channel and client mobility. We address this later in Section 4.1.4.

4.1.2 Packet Scheduling

Given the interference map of the network, the job of the MIM scheduler is to select an appropriate batch of packets from the queue, and prescribe their optimal order of transmission. Unfortunately, graph coloring algorithms on conflict graphs are not applicable in the case of MIM. This is because graph coloring assumes that conflicts between links are symmetric, whereas, due to the ordering constraints in MIM, link conflicts are actually *asymmetric*. This warrants new MIM-aware scheduling algorithms, that unsurprisingly, prove to be NP-hard. Therefore Shuffle employs a heuristic called **Least-Conflict Greedy**. In Least-Conflict Greedy, each packet in the queue is checked to identify its asymmetric conflicts with all other packets in the queue. Each packet is assigned a score based on such conflicts (a higher score if the packet *must* start first). Then, the scheduler runs a greedy ordering algorithm based on the scores of packets. The intuition is that packets with fewer conflicts (smaller score) will be included early in the batch, potentially accommodating more concurrent links. Fairness and starvation issues are certainly important challenges.

4.1.3 Staggered Transmissions

The scheduler outputs a batch of packets, and their staggered order of transmission. Each packet from this batch is forwarded to the corresponding AP, along with its *duration of stagger*. The APs begin transmission based on their prescribed stagger, illustrated in Figure 6. Notice that the transmissions are staggered in the order AP1→C13 before AP3→C32 before AP2→C21. Acknowledgments to these transmissions may or may not be MIM-aware. To support MIM-aware ACKs, the "stagger duration" of ACKs can be piggybacked in the data packets. To reduce clock synchronization issues, the stagger duration can be specified as the time between the receipt of the data and the start of the ACK transmission. Clients transmit the ACKs after this specified wait. Transmission failures are handled through scheduled retransmissions from the controller. The failed packet's priority is increased to ensure quicker scheduling.

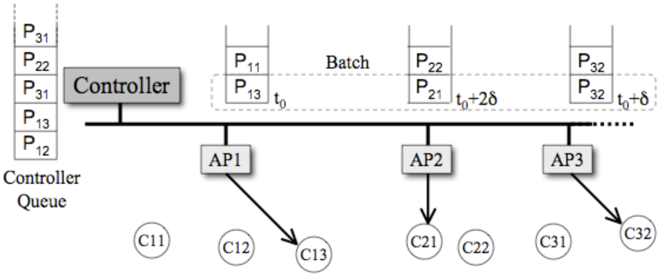

Figure 6: Batch transmission with suitable stagger.

4.1.4 Handling Channel Fluctuations

A rehearsal produces a snapshot of the interference relationships in the network. However, the interference relations change over time, and scheduling algorithms must remain abreast of such changes. For this, Shuffle employs continuous *opportunistic rehearsals*. The basic idea with opportunistic rehearsal is that clients and APs continuously record RSSI values of ongoing transmissions, and time-stamp them. These $< RSSI,\ time >$ tuples from the recent past are piggybacked in ACKs or other packets that clients send to APs. The APs forward the clients' (and their own) tuples to the controller, which in turn correlates them over time to refresh the interference map. Scheduling decisions are based on this frequently refreshed interference map, allowing Shuffle to cope with fading and fluctuations. Our measurements have shown that the conflict graph of the network remains stable over hundreds of packets, giving us reason to believe that opportunistic rehearsal will be fast enough to cope with channel changes. Of course, this needs to be confirmed with full scale implementation.

4.2 Limitations, Issues, and Extensions

While time synchronization is necessary to stagger packets, we believe that it need not be too tight. This is because we require packets to be staggered *more* than the preamble of an earlier packet. Conservative staggering (by adding a factor-of-safety more to the preamble duration) can accommodate clock skews. The impact on performance may only be marginal.

Shuffle needs to account for upload traffic. For this, clients could express their intent by setting a flag on ACK packets. On getting a flagged ACK, the controller could schedule the client in the next batch of transmissions. For clients that do not have an ACK to send, a short periodic time window can be allocated for contention-based (CSMA) transmissions.

Interferences in the 2.4GHz band, such as from microwaves, cordless phones, or even from other devices near the periphery of the EWLAN, can affect the scheduled transmissions. Shuffle can account for them through rehearsal if they persist for long durations. If they are transient, Shuffle will rely on retransmissions to cope with them.

The discussion of Shuffle thus far assumes that all transmissions use a fixed rate and power. If APs are allowed to transmit at varying rates and power levels, the controller may be able to extract higher spatial reuse. The impact of rate and power on MIM-aware schedules is part of our future work.

5. EVALUATION

We evaluate Shuffle using the Qualnet simulator. MIM capabilities were carefully modeled into the PHY and MAC layer of the simulator. The EWLAN controller was assumed to have a processing latency of 50μs, and the wired backbone was assigned 1 Gbps data rate. We used 802.11a with transmission power 19dBm, two ray prop-

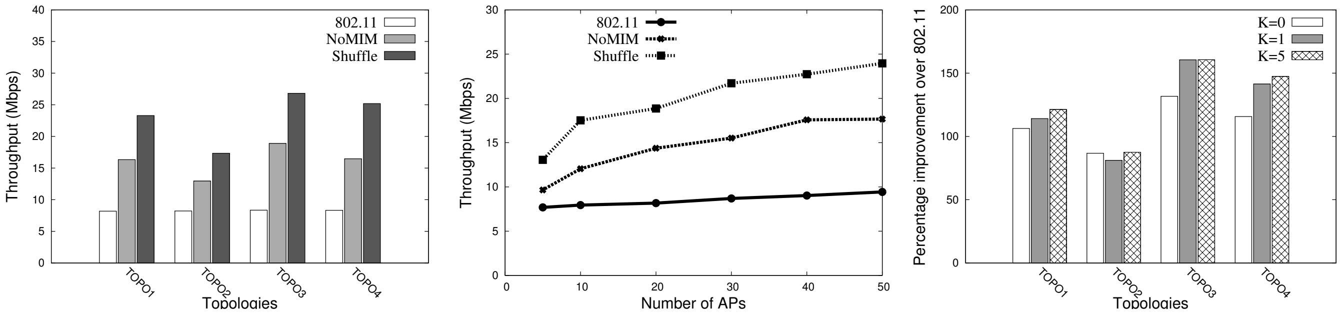

Figure 7: (a) Throughput for university topologies. Shuffle outperforms NoMIM and 802.11, characterizing the gains from MIM. (b) Higher AP density creates more opportunities for concurrency with MIM. (c) Percentage throughput improvement with channel fading – Shuffle performs well under Rayleigh and Ricean fading.

agation model, transmission rate 12Mbps, and a PHY layer preamble duration of 20μs. While we evaluated latency, fairness, and some scheduling variants, we report results for the primary metric of throughput.

We compare Shuffle with 802.11 and an MIM-incapable scheme, called *NoMIM*. The gain from MIM alone is reflected in the difference between *Shuffle* and *NoMIM*. The difference between *NoMIM* and *802.11* characterizes the gains from centralized scheduling. Fig. 7(a) presents throughput comparisons for topologies taken from university buildings with different number of APs on the same channel; each AP was associated to around 6 clients. As a special case, the second topology has APs associated to 20 clients, resembling a classroom setting. Shuffle consistently outperforms NoMIM and 802.11, confirming the potential of MIM-aware reordering. Evident from the difference between NoMIM and 802.11, a fair amount of benefit is also available through centralized scheduling in EWLANs.

Impact of AP density

Next generation EWLANs may be envisioned to have very high density of access points (perhaps each Ethernet-capable desktop will act as an AP). To understand Shuffle's scalability in high density environments, we randomly generated topologies in an area of 100x150 m^2. We placed an increasing number of APs (ranging from 5 to 50) at uniformly random locations in this region. Each AP is associated with 4 clients and the controller transmits CBR traffic at 1000 pkts/sec to each of the clients. Figure 7(b) illustrates that the throughput of shuffle increases as the density of APs increase. This is because the number of short (and hence high SINR) links increases with greater number of APs. This enables more links satisfying the SoI-Last threshold.

Impact of Fading

The above simulation results were obtained without channel fading, however, the impact of channel fading can be severe, and the protocol needs to adapt to it over time. To evaluate our opportunistic rehearsal mechanisms, we simulate Ricean fading with varying K factors, and log-normal shadowing. Figure 7(c) shows the percentage improvement of Shuffle over 802.11 for different values of K. For $K = 0$ (Rayleigh Fading), the fading is severe and the improvements are less than at higher values of K. Still, the improvements are considerable, indicating Shuffle's ability to cope with time-varying channels. The improvements were verified to be a consequence of opportunistic rehearsals; when opportunistic rehearsal was disabled, the performance degraded.

6. SHUFFLE WITH MIM VS SIC

Successive interference cancellation (SIC) is a physical layer capability that can be exploited at the MAC layer to extract a weaker signal of interest (SoI) from a stronger interference [11]. This development calls into question the utility of Shuffle based on MIM. An obvious argument in favor of MIM is that it is feasible with current Atheros chipset based receivers whereas SIC capable receivers may be available sometime in future. Nevertheless, we address this issue in this section by first contrasting the abilities of MIM with SIC, and then arguing how Shuffle augments SIC towards even higher performance.

Suppose two frames S and I overlap at a receiver and assume S is the frame of interest. Conventionally S can be decoded only if it is stronger than I and arrives before I at the receiver. With MIM, S can be extracted regardless of whether S arrives before or after I. However, even MIM cannot help if S is weaker than I. Importantly, SIC empowers a receiver to decode I, subtract it from the combined signal (S + I + Noise), and then decode S from the residue (S + Noise). Of course, this description is an over-simplification for the purpose of brevity. Now, we observe that ordering of transmissions helps even with SIC. Consider the two cases when I is moderately stronger, or much stronger, than S. (1) If I is moderately stronger than S, then initiate I before S. This helps in decoding I first with a lower SINR threshold because its preamble is sent in the clear (recall SoI-first case from MIM). Once I is decoded well, it can be subtracted better, allowing better decoding of S. (2) However, if I is much stronger than S, then initiate I after S (recall SoI-last case from MIM). Since the receiver re-locks into I anyway, this ordering helps decode S's preamble in the clear, which is later beneficial for separating S from the residue. Thus, Shuffle's approach of reordering and staggering of transmissions facilitates concurrency in SIC as well. Now, in the case where the receiver is interested in both the frames, similar MIM-aware reordering can be helpful. One needs to view both S and I as signals of interest, say S1 and S2. Thereafter, the two signals need to be initiated in the appropriate order, depending on their relative strengths. Figure 8 shows this interplay of SIC and MIM through a logical representation.

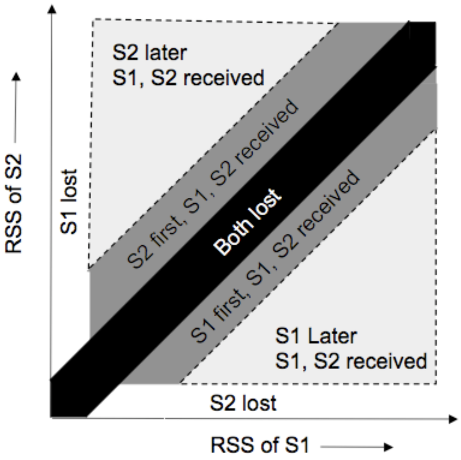

Figure 8: Ideal ordering of transmissions and corresponding reception outcomes (assuming S1 and S2 are not too weak). (1) when S1 and S2 of comparable strength, then both frames lost (black) (2) when S1 moderately stronger than S2, and S1 started before S2, both received (dark gray) (3) when S1 much stronger than S2, and S1 started after S2, both received (light gray). The upper square is symmetric.

SIC, when coupled with power control, opens up interesting possibilities for scheduling transmissions. Consider a scenario where S1 and S2 have packets to transmit to R1 and R2, respectively. Suppose the received signal strengths of S1 and S2 at R2 are comparable, i.e., fall within the middle black band in Figure 8. If they transmit concurrently, both will fail even with SIC because neither of the signals can be decoded for subsequent subtraction. On the contrary, if the S2 to R2 signal is made weaker by S2 transmitting at a lower power, i.e., moving from the middle black band of Figure 8 vertically towards x-axis, both transmissions can be successful. This is because both R1 and R2 can capture S1's frame which is relatively stronger, and R2 can then cancel S1's frame to extract S2's frame. The benefit of SIC is of course contingent on the signal strength values and in some cases may result in a lower rate for S2→R2.

An SIC-aware Shuffle system will also employ rehearsals and stagger transmissions as in Fig. 5. The only change is the input to the scheduler. Instead of MIM constraints, SIC will have to be satisfied while scheduling the batch of queued packets. SIC constraints permit higher spatial reuse than MIM constraints as they allow a weaker reception to be concurrent with a stronger interfering transmission. We plan to build and evaluate this order-sensitive link layer to exploit both MIM and SIC.

7. CONCLUSION

Physical layer capabilities like MIM and SIC are capable of better coping with interference. Although some benefits may be automatically available, significant improvements can be achieved if the link layer explicitly exploits these capabilities. This paper investigates link layer opportunities and challenges towards harnessing these capabilities. We present an MIM-aware framework named *Shuffle* that reorders transmissions to enable concurrent communications. Theoretical and simulation studies show that building MIM-aware networks are worthwhile; our small-scale prototype validates its practicality. Based on these results, we are implementing *Shuffle* with the aim of providing a fully functional system solution.

8. REFERENCES

[1] NS2, "http://www.isi.edu/nsnam/ns/," .
[2] Scalable Network Technologies, "Qualnet v2.6.1," www.scalable-networks.com.
[3] A. Kochut, A. Vasan, A. U. Shankar, and A. Agrawala, "Sniffing out the correct physical layer capture model in 802.11b," in *ICNP*, 2004.
[4] T. Nadeem and L. Ji, "Location-Aware IEEE 802.11 for Spatial Reuse Enhancement," *IEEE Trans. Mobile Computing*, vol. 6, no. 10, Oct. 2007.
[5] J. Lee et al, "An Experimental Study on the Capture Effect in 802.11a Networks," in *WinTECH*, 2007.
[6] Mythili Vutukuru, Kyle Jamieson, and Hari Balakrishnan, "Harnessing Exposed Terminals inWireless Networks," in *NSDI*, 2008.
[7] P. Bahl et al., "DAIR: A Framework for Managing Enterprise Wireless Networks Using Desktop Infrastructure," in *HotNets*, 2005.
[8] N. Ahmed et al., "Interference mitigation in enterprise WLANs through speculative scheduling," in *Mobicom*, 2007.
[9] R. Maheshwari, S. Jain, and S. R. Das, "On Estimating Joint Interference for Concurrent Packet Transmissions in Low Power Wireless Networks," in *WinTECH*, 2008.
[10] H. Chang, V. Misra, and D. Rubentein, "A General Model and Analysis of Physical Layer Capture in 802.11 Networks," *INFOCOM*, 2007.
[11] D. Halperin et al., "Taking the Sting out of Carrier Sense: Interference Cancellation for Wireless LANs," in *MOBICOM*, 2008.