Generalizing to Different Numbers of Antennas at the Transmitter and Receiver: Finally, n+ generalizes to scenarios where a transmitter and its receivers have different numbers of antennas. Consider, for example, the scenario in Fig. 4 where a 2-antenna access point, AP1, has a single-antenna client, c1, and a 3-antenna AP, AP2, has two 2-antenna clients, c2 and c3. Say that the single-antenna client is transmitting to its AP. In today's networks, this will prevent any other node from transmitting concurrently. However, with n+, the 3-antenna AP can transmit concurrently two packets, one to each of its clients, i.e., p_2 to client c2 and p_3 to client c3 as shown in Fig. 4.

So how does the 3-antenna AP transmit these concurrent packets while protecting the ongoing reception at the 2-antenna AP? To protect the ongoing reception, the 3-antenna AP must ensure that both of its transmitted packets (p_2 and p_3) are received at the 2-antenna AP along a direction orthogonal to the signal of interest to that AP, i.e., the signal from c1 (called p_1 in Fig. 4). This allows the 2-antenna AP to continue to receive its client's signal without interference, as shown in the bottom graphs (below AP1) in Fig. 4. The 3-antenna AP also needs to ensure that its transmission to one client does not create interference at the other client. Since each of its clients has two antennas and hence receives signals in a 2-dimensional space, this goal can be achieved if the 3-antenna AP ensures that each client receives the unwanted signal aligned along the interference it already sees from the ongoing transmission of the single-antenna client, (i.e., along p_1), as shown in the bottom graphs in Fig. 4.

3. n+'S DESIGN

n+ is a random access protocol that enables nodes with any number of antennas to contend for both time and degrees of freedom. It also has bitrate selection built-in.

3.1 Overview

Similar to 802.11, in n+, nodes listen on the wireless medium using carrier sense. If the channel is unoccupied, the nodes contend for the medium using 802.11's contention window and random backoff [5]. The node pair that wins the contention exchanges a light-weight RTS-CTS. The RTS-CTS allows nodes interested in contending for the remaining degrees of freedom to compute the channels to the receivers who won earlier contentions, in order to perform the required alignment or nulling. The RTS-CTS also includes the number of antennas that will be used in the transmission. After the RTS-CTS, the node pair proceeds to exchange the data packet followed by the ACK.

Unlike 802.11, n+ allows nodes who have more antennas than the current number of used degrees of freedom to contend for concurrent transmissions. The number of used degrees of freedom is equal to the number of ongoing transmissions, which a node can learn from prior RTS-CTS messages. As nodes contend for the unused degrees of freedom, they again use a contention window and random backoff similar to 802.11. However, while carrier sensing, nodes need to ignore the signals from past contention winners. To do so, n+ leverages that multi-antenna nodes receive the signal in a multi-dimensional space and, thus, can project on a space orthogonal to ongoing transmissions from past contention winners. Due to orthogonality, this space does not contain any interference from the ongoing transmissions, and thus, allows the nodes to perform carrier sense as if there were no ongoing transmissions. The process continues until all the degrees of freedom in the network have been used.

To illustrate how this design works, let us consider again the network in Fig. 3 which has three transmitter-receiver pairs. Each of the three transmitters carrier senses the medium and contends for the channel. Depending on who wins the contention, four different scenarios are possible. Fig. 5(a) shows the scenario where the 3-antenna pair, tx3-rx3, wins the contention and ends up using all three degrees of freedom. In this case, tx3 and rx3 exchange RTS-CTS, informing other nodes that they will use three degrees of freedom in their transmission. Since the other two transmitters have fewer than three antennas, they cannot support any additional degrees of freedom, and hence stop contending until the end of this transmission.

In the second scenario shown in Fig. 5(b), the two-antenna pair, tx2-rx2, wins the contention and uses two degrees of freedom. The first transmitter, tx1, notices that the channel is occupied and drops out of contention since it has only a single antenna. The third transmitter, tx3, on the other hand, has three antennas and therefore can deliver an additional packet. So it contends for the medium and wins the third degree of freedom. Since tx3 must not interfere with the ongoing transmission of tx2-rx2, it nulls its signal on the two antennas at rx2. This consumes two antennas at tx3, leaving it one antenna to transmit one stream to its own receiver, rx3.

The third scenario in Fig. 5(c) occurs when tx1-rx1 wins the contention. Since only a single degree of freedom is used, both tx2 and tx3 contend for the remaining two degrees of freedom. If tx3 wins, it needs to use one of its antennas to null its signal at rx1, which leaves it two antennas to send two concurrent streams to its own receiver, rx3, as in Fig. 5(c).

The last scenario shown in Fig. 5(d) occurs when the nodes win contention in the following order: tx1-rx1, tx2-rx2, tx3-rx3. It is similar to the example described in §2, where each of the pairs ends up transmitting a single packet.

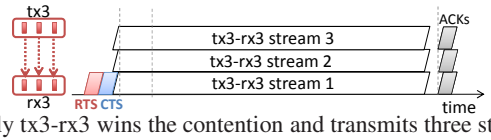
Finally, a few additional points are worth noting:

- n+ makes a node that joins ongoing transmissions end its transmission at about the same time as prior transmissions, which it learns from their light-weight RTS-CTS exchange. This design choice forces the medium to become idle at the end of each joint transmission, and hence prevents starving nodes that have only one antenna. Requiring all nodes to end their concurrent transmissions with the first contention winner means that nodes may need to fragment or aggregate packets. Various link layer protocols require packet fragmentation or aggregation. For example, 802.11n requires the driver to be able to aggregate multiple packets to create an aggregate frame [6], whereas old ATM networks require packet fragmentation [17]. n+ leverages these methods.
- Instead of sending the ACKs one after the other, the receivers transmit their ACKs concurrently. These concurrent transmissions are analogous to the concurrent transmissions of the data packets, and can be achieved using a combination of nulling and alignment (see §3.3).

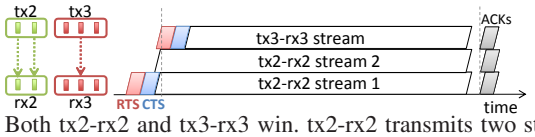
The above provides an overview of n+. The next few sections explain how we realize this design. We first develop the details of the algorithms and the system architecture, and leave addressing the practical issues until §4.

3.2 Carrier Sense Despite Ongoing Transmissions

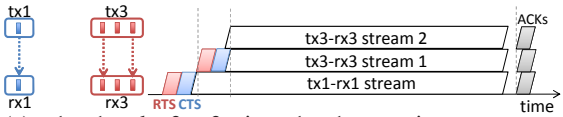
In n+, nodes use 802.11's carrier sense to contend for additional concurrent transmissions, even after some nodes have already won earlier contention rounds and started their transmissions. For this approach to work effectively, carrier sense should be oblivious to the ongoing transmissions. n+ satisfies this constraint as follows: In n+, a node that is interested in sensing the medium first computes the channel for the ongoing transmissions (which it does using the



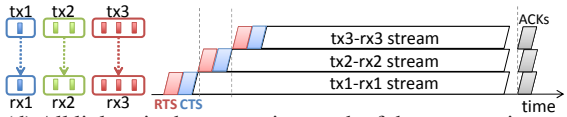
(a) Only tx3-rx3 wins the contention and transmits three streams.



(b) Both tx2-rx2 and tx3-rx3 win. tx2-rx2 transmits two streams, and tx3-rx3 transmits one stream using the third degree of freedom.



(c) tx1-rx1 and tx3-rx3 win. tx1-rx1 transmits one stream, and tx3-rx3 transmits two streams using the remaining degrees of freedom.



(d) All links win the contention, each of them transmits one stream using one degree of freedom.

Figure 5—Medium access for the three link scenario

preamble in their RTS messages). These channels define a subspace where the ongoing transmissions live. If the node projects on a space orthogonal to this subspace (using standard algebra [23]), the node will see no signal from ongoing transmissions, and hence can perform standard 802.11 carrier sense.

We name this approach multi-dimensional carrier sense. To illustrate how it works, consider again the example in Fig. 3, where we have three pairs of nodes: a single-antenna pair tx1-rx1, a 2-antenna pair tx2-rx2, and a 3-antenna pair tx3-rx3. Let us focus on the 3-antenna transmitter, tx3, as it senses the medium.

Say the single-antenna transmitter, tx1, wins the first round of contention and is already transmitting some signal p, hence using the first degree of freedom. Say tx3 wants to contend for the second degree of freedom. tx3 should sense the medium, but ignore the signal p from tx1. To do so, tx3 first computes the channel from tx1 to its three antennas using the preamble in tx1's RTS. We refer to these channels as h_1, h_2, and h_3. Since tx3 has three antennas, the received signal lies in a 3-dimensional space and can be written as:

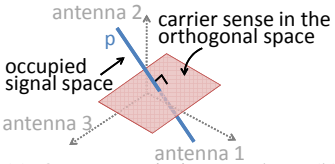
$$\vec{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} p = \vec{h}_{tx1} p,$$

where $\vec{h}_{tx1}$ is the channel vector $[h_1, h_2, h_3]^T$. Thus, for different symbols p transmitted by tx1, the received signal at tx3 changes over time, but merely moves along the one-dimensional vector $\vec{h}_{tx1}$, shown in Fig. 6(a). Therefore, by projecting on the 2-dimensional subspace orthogonal to this vector, (the red plane in Fig. 6(a)), tx3 eliminates interference from tx1 and can carrier sense for the remaining degrees of freedom. Since a 2-dimensional subspace is defined by any two distinct vectors in it, tx3 can project on the subspace orthogonal to p by simply picking any two vectors in the subspace, e.g., $\vec{w}_1$ and $\vec{w}_2$, and projecting on them to get:

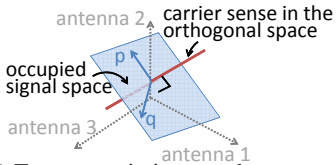
$$\vec{y}' = \begin{pmatrix} \vec{w}_1 \cdot \vec{y} \\ \vec{w}_2 \cdot \vec{y} \end{pmatrix},$$

where $\cdot$ denotes the dot product operation. If tx1's signal, p, is the only ongoing transmission, then $\vec{y} = \vec{h}_{tx1} p$, and by definition of orthogonality, $\vec{y}' = \vec{0}$. Thus, if tx3 performs carrier sense by sensing the signal after projection, $\vec{y}'$, it sees that the second degree of freedom is still unoccupied.

Now, say transmitter tx2 wins the second degree of freedom and starts transmitting its signal, q. Let h'_1, h'_2, and h'_3 be the channels from tx2 to tx3.[3] The three antennas at tx3 now receive the following combined signal from tx1 and tx2.



(a) One transmission on the medium



(b) Two transmissions on the medium

Figure 6—The received signal space as perceived by a 3-antenna node.

$$\vec{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} p + \begin{pmatrix} h'_1 \\ h'_2 \\ h'_3 \end{pmatrix} q = \vec{h}_{tx1} p + \vec{h}_{tx2} q,$$

where $\vec{h}_{tx2}$ is the channel vector for the second transmission. However, since tx3 is carrier sensing in the 2-dimensional space orthogonal to tx1's transmission, it computes:

$$\vec{y}' = \begin{pmatrix} \vec{w}_1 \cdot \vec{y} \\ \vec{w}_2 \cdot \vec{y} \end{pmatrix} = \begin{pmatrix} \vec{w}_1 \cdot \vec{h}_{tx1} p + \vec{w}_1 \cdot \vec{h}_{tx2} q \\ \vec{w}_2 \cdot \vec{h}_{tx1} p + \vec{w}_2 \cdot \vec{h}_{tx2} q \end{pmatrix} = \begin{pmatrix} \vec{w}_1 \cdot \vec{h}_{tx2} \\ \vec{w}_2 \cdot \vec{h}_{tx2} \end{pmatrix} q.$$

Thus, as opposed to the scenario in which only tx1 was transmitting and tx3 saw that the second degree of freedom is unused, tx3 sees that now $\vec{y}' \neq \vec{0}$, and hence the second degree of freedom is occupied.

Further, since the signal $\vec{y}'$ has no interference from tx1, and is equal to tx2's transmission, q, with a channel multiplier, tx3 can decode q using standard decoders. This allows tx3 to carrier sense not only by checking the power on the medium but also by cross correlating the preamble as in today's 802.11.

tx3 can use the same process to carrier sense and contend for the third degree of freedom. The only difference is that now it has to project on a space orthogonal to both tx1's and tx2's signals, as shown in Fig. 6(b). Thus, to summarize, for any number of concurrent transmissions the signal lives in a hyper-plane of the same dimension as the number of used degrees of freedom. To sense the medium, the node projects on the space orthogonal to the ongoing signals' hyper-plane, and performs carrier sense in this space.

3.3 Transmitting with Concurrent Transmissions

In n+, nodes that want to transmit in the presence of ongoing transmissions have to ensure that they do not interfere with those who already occupy the medium. This applies to the transmission of RTS, CTS, data, and ACK packets. In all of these cases, the approach is the same and relies on a combination of interference

[3] For ease of expression we lump the channels from tx2's two antennas into one term, i.e., $h'_1 = (h_{22} + h_{32}\alpha)$ in Eqs. 1a and 1b.

Term	Definition
K	number of ongoing streams/transmissions
M	number of antennas on a transmitter tx
N	number of antennas on a receiver rx
m	the maximum number of streams tx can transmit without interfering with the ongoing streams
n	the number of streams destined to rx, i.e., its **wanted streams**
$U, U^{\perp}$	the matrices defining the space of unwanted streams at rx and its orthogonal space
$\mathcal{R}, \mathcal{R}'$	receivers of ongoing streams and receivers of tx respectively
$\vec{v}_i$	the pre-coding vector of stream i

Table 1—Terms used in the description of the protocol.

nulling and alignment. For ease of exposition, we will describe it for the case of data packets.

(a) Definitions: Consider a scenario where there are K concurrent streams (i.e., K transmissions) on the medium. Let tx be an M-antenna transmitter that wants to transmit in the presence of the ongoing streams. Let m be the maximum number of concurrent streams that tx can transmit without interfering with the ongoing streams. For each stream that tx transmits, s_i, tx sends $\vec{v}_i s_i$, where $\vec{v}_i$ is an M-element *pre-coding* vector and each element v_{ij} describes the scaling factor for stream s_i transmitted from antenna j. Thus, the signal that tx transmits can be expressed as $\sum_1^m s_i \vec{v}_i$.

Let $\mathcal{R}$ be the set of receivers of the ongoing streams, and $\mathcal{R}'$ be the set of receivers of tx. Each receiver, rx, is interested in decoding the streams destined to itself, which we call the *wanted streams*. An N-antenna receiver, rx, that wants $n \leq N$ streams receives signals in an N-dimensional space, a subset of which is wanted and the rest is *the unwanted space*. We will use the matrix U to represent the unwanted space and $U^{\perp}$ to represent the space orthogonal to U. Table 1 summarizes our definitions.

(b) Protocol: The goal of our protocol is to compute the pre-coding vectors such that tx delivers its streams to its receivers without interfering with any of the ongoing streams. Our protocol proceeds in three steps as follows:

Step 1: Deciding whether to align or null. How does the transmitter, tx, decide whether to perform interference alignment or nulling at a particular receiver? The answer is simple. If the receiver has an unwanted space (i.e., $N > n$), it does not hurt to align the new interference in the unwanted space. However, if the wanted streams occupy the whole N-dimensional space in which rx receives signals, the transmitter has to null its interference at the receiver. Thus:

CLAIM 3.1 (WHERE TO NULL AND WHERE TO ALIGN). *To avoid interfering with the n wanted streams at an N-antenna receiver, rx, the transmitter nulls all of its streams at rx if $n = N$, and aligns its streams in rx's unwanted space, otherwise.*

Step 2: Computing the maximum number of concurrent streams that tx can transmit. The number of concurrent streams that tx can transmit is given by the following claim:

CLAIM 3.2 (NUMBER OF TRANSMITTED STREAMS). *A transmitter with M antennas can transmit as many as $m = M - K$ different streams concurrently without interfering with the ongoing K streams.*

The proof to this claim leverages the following two results:

CLAIM 3.3 (SATISFYING THE NULLING CONSTRAINT). *A transmitter can null its signal at an N-antenna receiver with n wanted streams (where $n = N$) by satisfying:*

$$\forall i = 1, \ldots, m, \qquad H_{N\times M}\vec{v}_i = \vec{0}_{n\times 1}, \tag{5}$$

where $H_{N\times M}$ is the channel matrix from tx to rx.

CLAIM 3.4 (SATISFYING THE ALIGNMENT CONSTRAINT). *A transmitter can align its signal in the unwanted space, U, of an N-antenna receiver with n wanted streams by satisfying:*

$$\forall i = 1, \ldots, m, \qquad U^{\perp}_{n\times N} H_{N\times M}\vec{v}_i = \vec{0}_{n\times 1}, \tag{6}$$

where $H_{N\times M}$ is the channel matrix from tx to rx.

The proofs of Claims 3.3 and 3.4 follow directly from the definitions of nulling and alignment. These two claims articulate the linear equations that tx's pre-coding vectors must satisfy. Eqs. 5 and 6 show that, independent of nulling or alignment, a receiver $\text{rx}_j \in \mathcal{R}$ that wants n_j streams results in a matrix equation of n_j rows. Hence, tx's pre-coding vectors have to satisfy a total number of linear equations equal to $\sum n_j$, where the sum is taken over the receivers in $\mathcal{R}$. This sum is simply the total number of ongoing streams K. Further, these equations are independent because of the independence of the channel matrices, the H's. Given that tx has M antennas and its pre-coding vectors have to satisfy K independent linear equations, there are exactly $M - K$ linearly independent such vectors. Thus, the number of different streams that tx can send is $m = M - K$.

Step 3: Computing the pre-coding vectors. Next, tx has to compute the pre-coding vectors. If tx has a single receiver, this task is fairly simple. tx combines the various nulling and alignment equations into one matrix equation as follows:

$$[H_1^T H_2^T \ldots (U_j^{\perp} H_j)^T \ldots]^T \vec{v} = \vec{0},$$

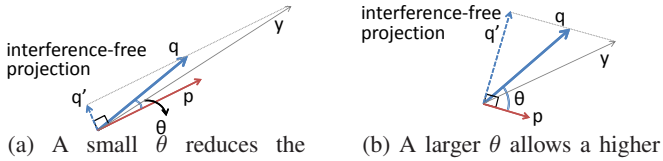
where $[.]^T$ is the matrix transpose. The solutions to this equation are the basis vectors of the null space of the matrix. Since the matrix dimensions are $K \times M$, there are $M - K$ such vectors.

If tx however has multiple receivers, as in Fig. 4, it needs to ensure that a stream that it sends to one receiver does not interfere with a stream that it sends to another receiver. For example, in Fig. 4, AP2 had to align the stream sent to each client in the unwanted space of the other client. This process however is similar to aligning at the receivers of ongoing streams expressed in claim 3.4. Specifically, say stream i is destined to receiver $\text{rx} \in \mathcal{R}'$. For every receiver $\text{rx}_j \in \mathcal{R}'$, different from rx, and whose unwanted space is U'_j, tx needs to ensure that $U_j'^{\perp} H'_j \vec{v}_i = \vec{0}$. Note that constraints for nulling or aligning at the receivers of ongoing streams are shared among all of tx's streams, whereas the constraints for nulling/aligning at tx's other receivers differ across tx's streams depending on the receiver of each stream. Combining all these constraints, tx can compute its pre-coding vectors as follows:

CLAIM 3.5 (COMPUTING THE CODING VECTORS). *Let $U^{\perp}_{n\times N}$ be the space orthogonal to the unwanted space at an N-antenna receiver, rx. For a receiver where the unwanted space is null, i.e., $n = N$, $U^{\perp}$ becomes the identity matrix, I. An M-antenna transmitter that wants to transmit m streams to receivers in $\mathcal{R}'$, while avoiding interference with receivers in $\mathcal{R}$, has to pick its coding vectors to satisfy:*

$$\begin{pmatrix} U_1^{\perp} H_1 \\ \vdots \\ U_{|\mathcal{R}|}^{\perp} H_{|\mathcal{R}|} \\ --- \\ U_1'^{\perp} H'_1 \\ \vdots \\ U'^{\perp}_{|\mathcal{R}'|} H'_{|\mathcal{R}'|} \end{pmatrix}_{M\times M} [\vec{v}_1 \ldots \vec{v}_m]_{M\times m} = \begin{pmatrix} 0\ldots 0 \\ \ddots \\ 0\ldots 0 \\ --- \\ I \end{pmatrix}_{M\times m}, \tag{7}$$

where $|.|$ is the cardinality of the set.



(a) A small θ reduces the achievable bitrate

(b) A larger θ allows a higher bitrate

Figure 7—The bitrate depends on the projection direction used to decode, and changes with the set of concurrent transmitters.

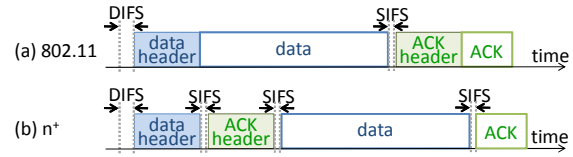


Figure 8—The Light-Weight RTS-CTS used in n$^+$: (a) a DATA-ACK exchange in 802.11n; (b) a DATA-ACK exchange in n$^+$, showing that n$^+$ does not send RTS-CTS, it rather separates the headers from the packets and sends all headers early on.

The proof follows directly from the discussion above. Thus, tx uses Eq. 7 to compute the pre-coding vectors. To do so, tx needs the channel matrices, H, which it obtains using reciprocity (as described in §2), and the alignment matrices, $U^{\perp}$, which are in the receivers' CTS messages. Once tx has the pre-coding vectors, it transmits its signal $\sum_1^m s_i\vec{v}_i$, which does not interfere with the wanted streams of any receiver.

3.4 Bitrate Selection

We discuss how a transmitter picks the best bitrate in the presence of ongoing transmissions. The challenge in this case is that bitrate selection has to be done on a per-packet basis because different packets share the channel with different sets of transmitters and hence require different bitrates. This constraint is very different from the standard assumptions made by today's bitrate selection algorithms, which use historical performance to predict the best bitrate.

We use a simple example to illustrate why the optimal bitrate of a MIMO node depends on concurrent transmitters. Consider a 2-antenna receiver that is interested in decoding a signal q in the presence of a concurrent transmission p. The 2-antenna receiver receives the combined signal y in a 2-dimensional space as shown in Fig. 7. To decode q, it uses the standard MIMO decoding algorithm called zero-forcing [32] to project the received signal y on a direction orthogonal to p. This projection removes all interference from p and yields a signal $q' = q\sin\theta$, where θ is the angle between the two signals p and q. The signal after projection is a scaled version of the original signal of interest and hence can be decoded using any standard decoder. The problem however is that, depending on the value of θ, the projected signal q' might have a large or small amplitude. A larger amplitude yields a higher SNR (signal-to-noise ratio) and hence a higher bitrate. A smaller amplitude yields a lower SNR and hence a lower bitrate.

In traditional MIMO systems where all concurrent streams/transmissions are from the same transmitter, p and q come from the same node and hence the angle between them does not change as long as the channels themselves do not change. However, when concurrent streams/transmissions are from different nodes, the angle changes from one packet to the next, as the set of concurrent transmitters changes, even if the channels themselves did not change. Thus, such a system requires a per-packet bitrate selection mechanism.

In n$^+$, each receiver uses the light-weight RTS of a packet to estimate the effective SNR (ESNR) after projection on the space orthogonal to ongoing transmissions. ESNR is a novel SNR-related metric that was recently proposed by Halperin et al [16]. Intuitively, the ESNR is similar to the SNR in that it captures the link quality; however, it is more useful for computing the best bitrate since it takes into account the impact of frequency selectivity. Given the ESNR, the receiver then chooses a valid bitrate using a table that maps ESNR to the optimal bitrate as shown by [16], and sends this decision back to the transmitter in the light-weight CTS message.

Note that a key characteristic of the above approach to bitrate selection is that a node can pick the optimal bitrate at the time it wins the contention without worrying about future contention winners. This is because transmitters that join ongoing transmissions avoid creating interference to existing receivers. This means that a single-antenna transmitter that wins the first degree of freedom observes a link quality that is unaffected by concurrent transmissions, and hence can use any standard bitrate selection algorithm to decide its best bitrate. A transmitter that wins contention in the presence of ongoing transmissions needs to pick the best bitrate given the current transmissions, but needs not worry about additional concurrent transmissions.

3.5 Light-Weight RTS-CTS

Before data exchange, n$^+$ needs the receiver to inform its sender of the best bitrate, and broadcast the alignment space to nodes that are interested in concurrent transmissions. This objective can be achieved by preceding each packet with an RTS-CTS handshake. RTS-CTS frames, however, would introduce a relatively high overhead. n$^+$ adopts a different design that achieves the goal but without sending any control frames. To do so, n$^+$ uses a recent design called the light-weight handshake, described in [20]. A light-weight handshake is based on the observation that 802.11 channel coefficients do not change for periods shorter than multiple milliseconds [32]. Hence, one can split a packet header from the packet body, and make the sender and receiver first exchange the data and ACK headers and then exchange the data and ACK bodies without additional headers. Fig. 8 compares this process with a standard data-ack exchange in 802.11.

The empirical study in [20] shows that the impact of separating a packet's header from its body is insignificant on decodability, namely the packet loss rate increases on average by 0.0005, which is negligible for a wireless network.

The overhead of a light-weight handshake is minimal. Specifically, the overhead is two SIFS intervals, as shown in Fig. 8, and a per header checksum. In addition, each protocol may augment the standard data or ACK header with protocol-specific fields. In the case of n$^+$, the standard data and ACK headers already contain most of the needed information. Specifically, they contain a preamble for computing the channels, the packet length which implies its duration given a bitrate, the number of antennas, and the sender and receiver MAC addresses. In addition, n$^+$ augments the ACK header with the bitrate and the alignment space. Since n$^+$ performs nulling and alignment on each OFDM subcarrier independently, a receiver needs to send the alignment space for each of the 802.11's 64 OFDM subcarriers. n$^+$ leverages that the channel coefficients change slowly with OFDM subcarriers [9], and hence the alignment space in consecutive subcarriers is fairly similar. Thus, n$^+$ sends the alignment space U of the first OFDM subcarrier, and the alignment difference $(U_i - U_{i-1})$ for all subsequent subcarriers. Our results from a testbed of USRP2 radios in both line-of-sight and non-line-of-sight locations (see Fig. 10) show that differential encoding can on average compress the alignment space into three OFDM symbols. Since the CRC and bitrate values fit within one

OFDM symbol, the header size in n+ increases by four OFDM symbols in the case of an ACK, and one OFDM symbol in the case of a data packet.

Thus, the total overhead from the light-weight handshake is 2 SIFS plus 4 OFDM symbols, which is about 4% overhead for a 1500-byte packet transmitted at 18 Mb/s. We note that these results are for USRP2 channels which have a 10 MHz width. 802.11 channels span 20 MHz and hence are likely to show more variability in the alignment space of different OFDM subcarrier. Hence, the number above should be taken as a rough estimate that indicates that the overhead is significantly smaller than the gain.

Finally, to support scenarios like the one in Fig. 4 where a single node transmits concurrently to multiple receivers, we allow a single light-weight RTS (i.e., the data header) to contain multiple receiver addresses along with the number of antennas used for each receiver. The receivers send their light-weight CTS's (i.e., their ACK headers), one after the other, in the same order they appear in the light-weight RTS.

4. PRACTICAL SYSTEM ISSUES

This section addresses a few practical issues.

Hidden Terminals and Decoding Errors: The light-weight handshake mechanism used by n+ has the side-effect of providing the functionality of RTS-CTS which alleviates the hidden terminal problem. Further, in n+, if a node misses or incorrectly decodes one of the RTS or CTS messages from prior contention winners or its own exchange, it does not transmit concurrently. Operationally this is similar to missing a traditional RTS or CTS.

Retransmissions: When an n+ node transmits a packet, it keeps the packet in its queue until the packet is acked. If the packet is not acked, the next time the node wins the contention, it considers the packet for transmission. However, since the node always needs to finish with other concurrent transmissions, the packet may be fragmented differently or aggregated with other packets for the same receiver.

Multipath: Our discussion has been focused on narrowband channels. However, the same description can be extended to work with wideband channels which exhibit multipath effects. Specifically, such channels use OFDM, which divides the bandwidth into orthogonal subcarriers and treats each of the subcarriers as if it was an independent narrowband channel. Our model naturally fits in this context. Specifically, like today's 802.11, n+ treats each OFDM subcarrier as a narrowband channel and performs nulling and alignment for each OFDM subcarrier separately.

Frequency Offset: To avoid inter-carrier interference, concurrent transmitters should have the same carrier frequency offset (CFO) with respect to every receiver. Thus, n+'s senders compensate for their frequency offset in a manner similar to that used in [28, 30]. Specifically, as they decode the RTS from the transmitter that won the first degree of freedom, all concurrent transmitters naturally estimate their frequency offset with respect to the first transmitter. They compensate for that frequency offset by multiplying their digital signal samples by $e^{j2\pi\Delta ft}$ where Δf is the frequency offset and t is time since the beginning of the transmission. This process synchronizes all transmitters in the frequency domain without requiring any explicit coordination.

Time Synchronization: To prevent inter-symbol interference (ISI), concurrent transmitters have to be synchronized within a cyclic prefix of an OFDM symbol [30]. To do this without any explicit coordination, n+ uses the technique in [30]. In particular, any transmitter that wants to join ongoing transmissions estimates the OFDM symbol boundaries of ongoing transmissions and synchronizes its transmission with them. To deal with additional delays due to channel propagation and hardware turn-around time, both the cyclic prefix and the OFDM FFT size are scaled by the same factor. A longer cyclic prefix provides additional leeway for synchronization at the transmitters, as shown in [30]. Further, this scaling does not increase the overhead because the percentage of cyclic prefix to data samples stays constant.

Imperfections in Nulling and Alignment: In practice, it is impossible to get perfect nulling or alignment due to hardware nonlinearities. This means that there is always some residual noise. The practical question however is: what level of residual noise is acceptable in these systems? The answer is: as long as the interference is reduced below the noise level of the hardware, the interference becomes negligible. For example, say that, in the absence of nulling or alignment, the interferer achieves a 25 dB SNR at a particular receiver. Then if nulling or alignment reduces the interference power by over 25 dB, the interference will be below the noise, and its impact is relatively negligible.

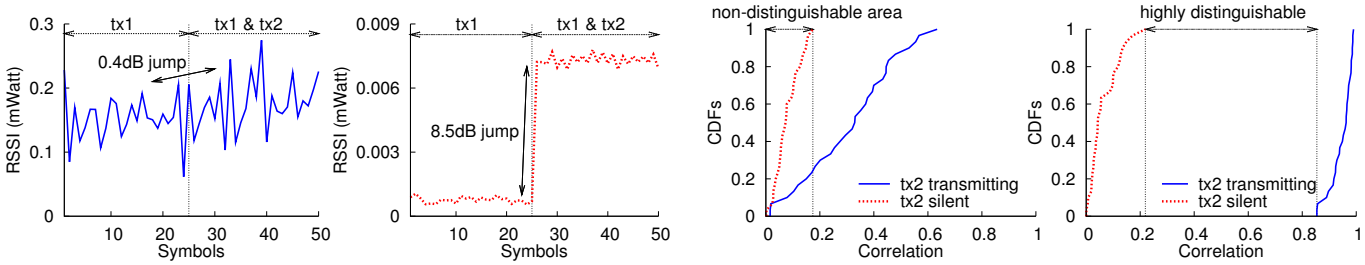
Thus, in n+ we make a transmitter join an ongoing transmission only if it can reduce its interference power below the noise power. Specifically, say that interference nulling and alignment in practice can reduce the transmitter power by L dB (our empirical results show that L is about 25–27 dB). A transmitter that wants to contend for the unused degrees of freedom estimates the power of its signal at each receiver of the ongoing transmissions. The transmitter can do so because it knows the channel to these receivers and hence it knows the attenuation its signal would experience. If the resulting signal power after channel attenuation is below L dB, the transmitter contends for transmitting concurrently. On the other hand, if the signal power after channel attenuation is still higher than L, the transmitter reduces its own transmission power so that after attenuation it is less than L dB. The transmitter contends (and if it wins the contention transmits) at this lower power, which can be canceled using practical interference nulling and/or alignment.

Complexity: Components used in n+ such as projections and estimation of the MIMO channel values are already used in current 802.11n for decoding point-to-point MIMO packets. Further, the computational requirement of computing the alignment and nulling spaces is similar to that of computing beamforming matrices in current 802.11n. Given the similarity between the components of n+ and those used in today's hardware, we believe that n+ can be built in hardware without significant additional complexity.

5. IMPLEMENTATION

We implement the design of n+ using software radios. Each node in the testbed is equipped with USRP2 boards [3] and RFX2400 daughterboards, and communicates on a 10 MHz channel. Since USRP2 boards cannot support multiple daughterboards, we build a MIMO node by combining multiple USRP2's using an external clock [2]. In our evaluation, we use MIMO nodes which have up to three antennas. Further, we build on the GNURadio OFDM code base, using different 802.11 modulations (BPSK, 4QAM, 16QAM, and 64QAM) and coding rates, to implement the effective-SNR based bitrate selection algorithm.

We implement the following components of our design: carrier sense, light-weight RTS-CTS, alignment and nulling, bitrate selection, and frequency offset correction. However, due to the timing constraints imposed by GNURadio, we evaluate carrier sense independently from light-weight RTS-CTS and data transmission. Also, we do not implement ACKs. To perform nulling and alignment efficiently, concurrent transmitters have to be synchronized within a



Power without projection | Power with projection | Correlation without projection | Correlation with projection

(a) Sensing Power **(b) Cross-correlation**

Figure 9—Performance of Carrier Sense in the Presence of Ongoing Transmissions. The figures show that projecting on a space orthogonal to the ongoing transmissions provides a high distinguishability between a particular degree of freedom being occupied or free.

Figure 10—The testbed. Dots refer to node locations.

cyclic prefix. To achieve this goal, we exploit USRP2 timestamps to synchronize the transmitters despite the delays introduced by operating in software. We send a trigger signal and make the transmitters log the time of detecting the trigger, t_{start}. The transmitters then add a large delay, t_Δ, and set the timestamps of their concurrent transmissions to $t_{start} + t_\Delta$. The value of t_Δ depends on the maximum delay due to software processing, which in our testbed is 5 ms.

6. RESULTS

We evaluate n$^+$ in the testbed environment shown in Fig. 10, and compare it against the existing 802.11n design.

6.1 Performance of n$^+$'s Carrier Sense

We start by examining the effect of projection on the performance of carrier sense in the presence of ongoing transmissions. 802.11's carrier sense has two components which together allow it to detect if the medium is occupied [18]. The first component checks whether the power on the medium is above a threshold. The second component cross-correlates 10 short OFDM symbols in the preamble to detect the presence of other 802.11 transmissions. We investigate how projecting on a space orthogonal to the ongoing transmissions affects these components.

Experiment: We focus on the example in Fig. 3, where there are three pairs of nodes, tx1-rx1, tx2-rx2, and tx3-rx3, which have 1, 2, and 3 antennas, respectively. We make tx3 sense the medium using the projection technique described in §3.2. tx1 starts transmitting followed by tx2. The timing between tx1 and tx2 is ensured by sending a trigger, logging the USRP timestamps when the node detected the trigger, and scheduling their transmissions with respect to the timestamp of the common trigger as detailed in §5. We log the signal at tx3 and process the logs offline to measure the channels and then project tx3's received signals on the space orthogonal to tx1. We repeat the experiment for different transmission powers of tx1 and tx2 to check that carrier sense works at low powers.

Results: First, we show in Fig. 9(a) an illustrated power profile at tx3, without and with projection. The graph on the left shows that if tx3 simply looks at the power on the medium without projecting, it might miss tx2's transmission because tx2's power is low in comparison with tx1's power. However, if tx3 projects on the space orthogonal to tx1, as in the graph on the right, it sees a relatively big jump in power when tx2 starts, and hence can more easily detect tx2's transmission.

Next, we show the result of cross correlating the preamble, without and with projection. We use the same size cross-correlation preamble as 802.11. We evaluate the system's ability to sense tx2's transmission in the presence of tx1's transmission. In this experiment, we focus on low SNR scenarios (SNR $<$ 3 dB) because sensing becomes harder when the sensed signal from tx2 has a low SNR.

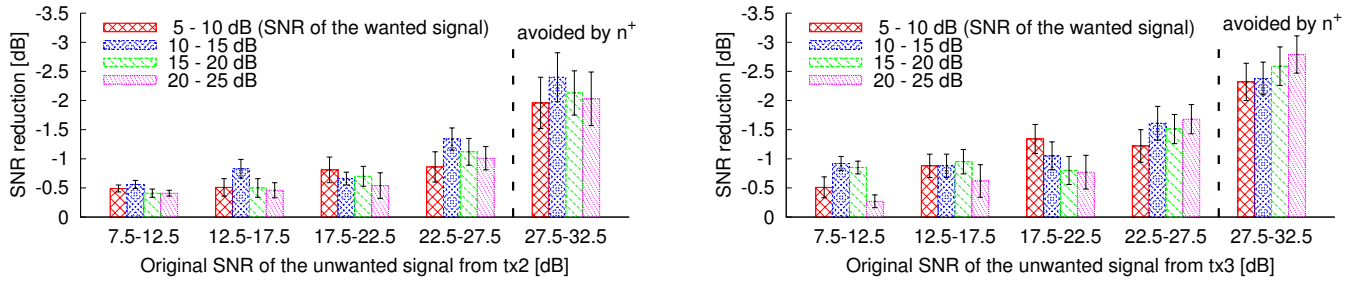
Fig. 9(b) plots the CDFs of the cross correlation values, without and with projection, both for the case of when tx2 is silent and transmitting. The figure shows that projecting on an orthogonal space (the graph on the right) provides a high distinguishability between the medium being unoccupied and occupied. This is because, with projection, the range of cross-correlation values measured when tx2 is silent is quite different from the cross-correlation values measured when tx2 is transmitting. In contrast, without projection (the graph on the left), about 18% of the cross-correlation values measured while tx2 is transmitting are not distinguishable from the case when tx2 is silent.

6.2 Performance of Nulling and Alignment

While in theory nulling and alignment can eliminate interference of unwanted transmissions, in practice, system noise and hardware nonlinearities lead to residual errors. Thus, we examine the accuracy of nulling and alignment in practice.

Experiment: To evaluate nulling, we use the scenario in Fig. 2, where a single-antenna pair tx1-rx1 and a 2-antenna pair tx2-rx2 transmit concurrently. The 2-antenna pair, tx2-rx2, nulls its signal at rx1 to avoid interfering with tx1's transmission. We randomly assign the four nodes, tx1, rx1, tx2, and rx2, to the marked locations in Fig. 10, and run the experiment in three phases: First, we make the link tx1-rx1 transmit alone to measure the SNR of the wanted traffic in the absence of the unwanted traffic. Second, we make the link tx2-rx1 transmit alone to measure the SNR of the unwanted traffic at rx1 in the absence of nulling. Third, we make tx1 and tx2 transmit concurrently and have tx2 null its signal at rx1. We measure the SNR of the wanted stream at rx1 after nulling, and compare it with its SNR in the absence of the unwanted stream. We repeat the experiment with different random locations in the testbed.

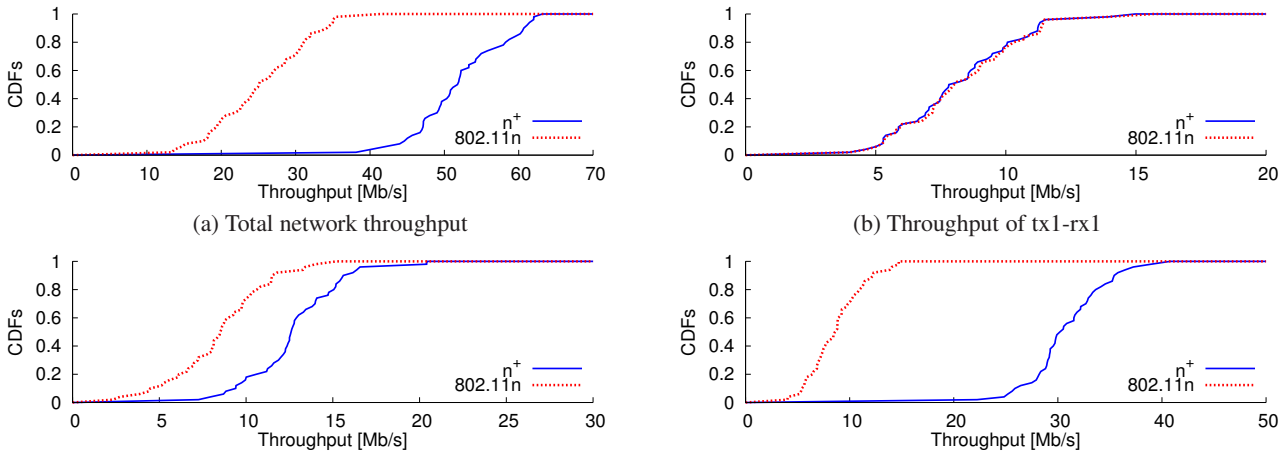
To evaluate alignment, we use the scenario in Fig. 3, i.e., we add a 3-antenna pair, tx3-rx3, to the two pairs, tx1-rx1 and tx2-rx2, used in the nulling experiment. As described in §2, the 3-antenna



(a) SNR reduction due to nulling

(b) SNR reduction due to alignment

Figure 11—Performance of Nulling and Alignment. The SNR loss of the wanted stream as a function of the original SNR of the unwanted streams in the absence of nulling or alignment. The figure shows that if the unwanted stream had a high SNR before nulling/alignment, it causes in a higher SNR loss for the wanted stream after nulling/alignment. Thus, n+ allows unwanted streams to transmit concurrently only if their original SNR is below 27 dB, which results in an average SNR loss of 0.8 dB for nulling and 1.3 dB for alignment.



(a) Total network throughput

(b) Throughput of tx1-rx1

(c) Throughput of tx2-rx2

(d) Throughput of tx3-rx3

Figure 12—Throughput Comparison. The figure plots the throughput obtained under n+ and the existing 802.11n design for the scenario in Fig. 3, where tx1-rx1 is a single-antenna node pair, tx2-rx2 is a 2-antenna node pair, and tx3-rx3 is a 3-antenna node pair.

pair, tx3-rx3, aligns its signal at rx2 along with the interference from tx1's transmission. Unlike the nulling experiment, the alignment experiment focuses on rx2, where the alignment is happening. Like the nulling experiment, however, it has three phases: First, tx1 and tx2 transmit concurrently, while tx3 stays silent to allow us to measure the SNR of the wanted stream at rx2, in the absence of interference from tx3. Second, tx3 transmits to rx2 alone to measure the SNR of the unwanted traffic in the absence of alignment. Last, all three transmitters transmit concurrently, and tx3 aligns its signal with that of tx1, as described in §2. We measure the difference in the SNR of rx2's wanted stream without interference and with interference alignment. We repeat the experiment with different random assignment of nodes to locations in Fig. 10.

Results: Fig. 11 plots the difference in the SNR of the wanted stream due to the presence of the unwanted stream, after nulling and alignment. The SNR difference is plotted as a function of the SNR of the unwanted (i.e., interfering) stream. Different bars refer to different SNRs of the wanted stream. The figure reveals four main points.

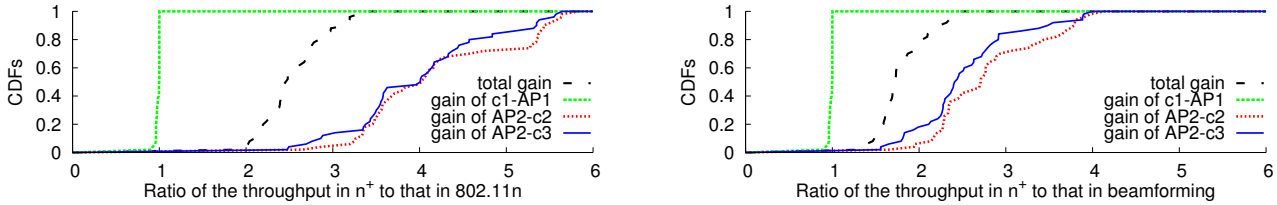
- When the power of the unwanted stream without nulling or alignment is in the range $[7.5, 32.5]$ dB, nulling and alignment reduce the impact of interference on the wanted signal to $[0.5, 3]$ dB.
- The residual interference after nulling or alignment depends on the original SNR of the unwanted signal before nulling or alignment. Thus, n+ takes this issue into account, and forces a node that wants to join ongoing transmissions to lower its interference power below a threshold $L = 27$ *dB*, as marked in the figure.
- Given n+'s threshold, the average interference power after nulling is 0.8 dB, and after alignment is 1.3 dB.
- Nulling has a lower residual error than alignment. This is because nulling requires estimating only the channel from the interfering transmitter to the receiver. Alignment, on the other hand, also requires estimating the unwanted subspace at the receiver. Since the latter estimation adds additional noise, alignment is less accurate than nulling.

6.3 Throughput Comparison

Next, we investigate the impact of nulling and alignment on throughput. We also compare the throughput obtained with n+ to that obtained with the existing 802.11n.

Experiment: Again, we consider the scenario in Fig. 3, which has three node pairs: tx1-rx1, tx2-rx2, and tx3-rx3, which have 1, 2, and 3 antennas respectively. Each run consists of a different assignment of nodes to locations in Fig. 10. The choice of which nodes win the contention is done by randomly picking winners. For 802.11n, an M-antenna node that wins the contention transmits an 1500 byte packet to its receiver using M concurrent streams. Similarly, for n+, an M-antenna node that wins the first contention transmits an 1500 byte packet using M streams. Latter contention winners in n+ end their transmissions at the same time as the first contention winner.

For both 802.11n and n+, the bitrate is chosen according to the



(a) Throughput gain in comparison with 802.11n

(b) Throughput gain in comparison with beamforming

Figure 13—Throughput gain. For the scenario in Fig. 4 where the transmitter and receiver have a different number of antennas, n+ provides an average network throughput improvement of 2.4x over 802.11n and 1.8x over beamforming.

algorithm in [16], which maps the effective SNR to the optimal bitrate. In n+ each node picks its bitrate at the time of joining the concurrent transmission, independently of later contention outcomes. To achieve this behavior with GNURadios, the RTS-CTS messages are sent in a staggered fashion. For example, if the three pairs are each transmitting one stream, then tx1 first transmits its RTS message alone which is used by rx1 to compute its bitrate. Next, tx2 sends its RTS message in the presence of tx1's transmission, which rx2 uses to compute the bitrate for tx2. To compute the bitrate to be used by tx3, tx3 sends its RTS message in the presence of transmissions from both tx1 and tx2. Finally, tx1, tx2 and tx3 use the bitrates picked above to transmit their data packets concurrently.

Results: Fig. 12 plots the CDFs of the throughputs of each pair and the total throughput, both under n+ and 802.11n. The CDFs are taken over different locations. They show:

- On average, the total network throughput doubles when the nodes use n+, as opposed to current 802.11n.
- Nodes with multiple antennas achieve significant throughput gains, and the gains increase with increased number of antennas. Specifically, the 2-antenna pair experiences an average throughput gain of 1.5x, and the 3-antenna pair experiences an average throughput gain of 3.5x.
- There are three reasons for these throughput gains. First, n+ allows the nodes to transmit three streams at any point in time, providing a multiplexing gain. Second, in 802.11n, each transmitter is given an equal chance to transmit a packet. Thus, nodes that have low throughput occupy more time on the medium than nodes with high throughput. Since single-antenna nodes typically have lower throughput as compared to multi-antenna nodes, which can transmit concurrent streams, the network throughput of existing 802.11n is bottlenecked by single-antenna nodes. In contrast, in n+, since multi-antenna transmitters can join the ongoing transmission of single-antenna nodes, it does not matter that the single-antenna nodes take too much time on the medium. Third, allowing multiple nodes to transmit simultaneously provides a power gain [32]. In particular, the FCC limits the maximum transmission power of a single transmitter. Hence, 802.11n is limited by the amount of power on any single transmitter. In contrast, having multiple nodes transmit simultaneously (as in n+) increases the total power on the medium, which increases the capacity of the system.
- The reduction in throughput at the single-antenna node is less than 3%. This number is fairly small because the residual interference after nulling is 0.8 dB (as shown above), which has a negligible impact on throughput.
- While the residual interference after alignment is slightly higher than nulling (1.3 dB), it does not translate to a throughput loss. This is because the 2-antenna receiver, where alignment occurs, gains more from concurrent transmissions than the small loss due to alignment errors.

6.4 Performance with a Different Number of Transmit and Receive Antennas

We check that n+ increases the throughput even when the transmitter and receiver have a different number of antennas.

Experiment: We repeat the throughput evaluation experiment described in the last section but consider the scenario shown in Fig. 4 where a 2-antenna access point, AP1, is receiving from a single-antenna client, c1, and a 3-antenna AP, AP2, is transmitting to two 2-antenna clients. The rest of the setup mirrors that of the previous experiment. In addition to comparing the throughput of n+ with 802.11n, we also compare it with prior work on beamforming [7] which can be applied to this scenario (but does not apply to the previous one). In particular, when the 3-antenna AP wins contention, beamforming allows it to transmit three streams concurrently, two to one client and one to the other.

Result: For space limitation we plot the CDFs of n+'s throughput gains in comparison with 802.11n in Fig. 13(a) and beamforming [7] in Fig. 13(b), i.e., the ratio of the throughput in n+ to that in 802.11n and beamforming, respectively. The figures show that the total network throughput increases by 2.4x and 1.8x over 802.11n and beamforming. Further, on average, the single-antenna client experiences a negligible reduction in throughput of 3.2%, whereas the other two clients experience a throughput gain of 3.5-3.6x over 802.11n and 2.5-2.6x over beamforming. As before, the large gains over 802.11n and beamforming are due to both having more concurrent transmissions and providing multi-antenna nodes more opportunities to transmit concurrently with the single-antenna transmitter. These results show that n+ extends to scenarios where the transmitter and the receiver have a different number of antennas.

7. RELATED WORK

In the last few years, MIMO networks have attracted much attention from both the theoretical and empirical research communities. This resulted in new powerful theories including virtual MIMO and interference alignment [8, 22, 19] and led to pioneering systems that expanded and validated the theory [13, 31, 7]. Our work builds on this foundation to provide the first random access MIMO protocol where nodes contend for both degrees of freedom and medium time and pick the best bitrate for their transmissions in a fully distributed manner without a centralized coordinator.

Our work is mostly related to recent empirical work on MIMO systems [7, 31, 13]. Past systems however require concurrent transmissions to be pre-coded together at a single transmitter (as in beamforming [7]), decoded together at a single receiver (as in SAM [31]), or the transmitters or the receivers have to be controlled over the Ethernet by a single master node (as in IAC [13]). In contrast, n+ does not require a centralized coordinator.

Our work on carrier sense in the presence of ongoing transmissions is related to packet detection in ZigZag [12] and carrier counting in SAM [31]. These schemes detect the number of concurrent

transmissions using preamble correlation. In contrast, n+ projects on a space orthogonal to ongoing transmissions, which cancels out the interference.

n+ also builds on prior theoretical work on MIMO [29, 27]. The most relevant theoretical work to this paper is past work on multi-user MIMO, interference alignment and cognitive MIMO. Multi-user MIMO allows multiple clients to be served simultaneously by a single base station that has more antennas than any of the clients. A number of techniques, such as beamforming, dirty paper coding, linear decorrelators, and successive interference cancellation, have been proposed to achieve the capacity of both the uplink and the downlink [11, 21, 24]. More recently, work on interference alignment [8, 19, 22] has shown new capacity results for multi-user MIMO channels. Finally, theoretical work on cognitive MIMO [10, 26, 25] has advocated the use of MIMO to ensure that secondary users can coexist with the primary users, without creating interference to the primary users. These papers provide only theoretical solutions and typically target specific topologies. n+ builds on this foundational work but differs from it in that it focuses on a random access protocol where nodes do not have to coordinate explicitly before they access the medium. Furthermore, n+ is implemented and shown to work in actual wireless networks.

8. CONCLUSION

This paper presents n+, a random access protocol for heterogeneous MIMO networks. We analytically show that n+ can always use as many degrees of freedom as the transmitter with the maximum number of antennas while maintaining the random access property of 802.11. We show via a prototype implementation that n+ can significantly improve the network throughput. We believe that as the diversity in the size and processing power of devices increases, n+ is well positioned to exploit these differences and provide a better utilization of network resources.

Acknowledgments: We thank Nabeel Ahmed, Arthur Berger, Haitham Hassanieh, Nate Kushman, Hariharan Rahul, Lenin Ravindranath, Mythili Vutukuru, and Richard Yang for their insightful comments. This research is supported by DARPA ITMANET, NSC and NSF.

9. REFERENCES

[1] 4x4 MIMO Technology. http://www.quantenna.com/4x4-mimo.html.
[2] ACQUITEK Inc., Fury GPS Disciplined Frequency Standard. http://www.acquitek.com/fury/.
[3] Ettus Inc., Universal Software Radio Peripheral. http://ettus.com.
[4] System Description and Operating Principles for High Throughput Enhancements to 802.11. IEEE 802.11-04/0870r, 2004.
[5] *IEEE Std 802.11-1997*, pages i –445, 1997.
[6] *IEEE Std 802.11n-2009*, pages c1 –502, 2009.
[7] E. Aryafar, N. Anand, T. Salonidis, and E. W. Knightly. Design and Experimental Evaluation of Multi-User Beamforming in Wireless LANs. In *ACM MobiCom*, 2010.
[8] V. Cadambe and S. Jafar. Interference Alignment and Degrees of Freedom of the K-User Interference Channel. *IEEE Trans. Inf. Theory*, 54(8):3425 –3441, 2008.
[9] O. Edfors, M. Sandell, J. Van de Beek, S. Wilson, and P. Borjesson. OFDM Channel Estimation by Singular Value Decomposition. *IEEE Trans. Comm.*, 46(7):931–939, 2002.
[10] S. Ganesan, M. Sellathurai, and T. Ratnarajah. Opportunistic Interference Projection in Cognitive MIMO Radio with Multiuser Diversity. In *IEEE DySPAN*, 2010.
[11] G.J.Foschini. Layered Space-Time Architecture for Wireless Communication in a Fading Environment When Using Multi-Element Antennas. In *Bell LAbs Technical Journal*, 1996.
[12] S. Gollakota and D. Katabi. Zigzag Decoding: Combating Hidden Terminals in Wireless Networks. In *ACM SIGCOMM*, 2008.
[13] S. Gollakota, S. D. Perli, and D. Katabi. Interference Alignment and Cancellation. In *ACM SIGCOMM*, 2009.
[14] M. Guillaud, D. Slock, and R. Knoop. A Practical Method For Wireless Channel Reciprocity Exploitation Through Relative Calibration. In *ISSPA*, 2005.
[15] M. Guillaud, D. Slock, and R. Knopp. A Practical Method for Wireless Channel Reciprocity Exploitation through Relative Calibration. In *Signal Processing and Its Applications*, 2005.
[16] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Predictable 802.11 Packet Delivery from Wireless Channel Measurements. In *ACM SIGCOMM*, 2010.
[17] R. Handel and M. Huber. *Integrated Broadband Networks; An Introduction to ATM-Based Networks*. Addison-Wesley Longman Publishing Co., Inc., 1991.
[18] J. Heiskala and J. Terry. *OFDM Wireless LANs: A Theoretical and Practical Guide*. Sams Indianapolis, IN, USA, 2001.
[19] S. Jafar and S. Shamai. Degrees of Freedom Region of the MIMO X Channel. *IEEE Trans. Inf. Theory*, 54(1):151–170, 2008.
[20] K. C.-J. Lin, Y. Chuang, and D. Katabi. A Light-Weight Wireless Handshake. In *MIT Tech Report, 2011*.
[21] R. Lupas and S. Verdu. Linear Multiuser Detectors for Synchronous Code-Division Multiple-Access Channels. *IEEE Trans. Inf. Theory*, 35(1):123–136, Jan. 1989.
[22] M. Maddah-Ali, A. Motahari, and A. Khandani. Communication over MIMO X Channels: Interference Alignment, Decomposition, and Performance Analysis. *IEEE Trans. Inf. Theory*, 54(8):3457–3470, 2008.
[23] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2001.
[24] M.K.Varanasi and T.Guess. Optimum Decision Feedback Multiuser Equalization and Successive Decoding Achieves the Total Capacity of the Gaussian Multiple-Access Channel. In *Proceedings of the Asilomar Conference on Signals, Systems and Computers*, 1997.
[25] B. Nosrat-Makouei, J. Andrews, and R. Heath. User Admission in MIMO interference Alignment Networks. In *IEEE ICASSP, Prague, May 2011.*
[26] S. Perlaza, N. Fawaz, S. Lasaulce, and M. Debbah. From Spectrum Pooling to Space Pooling: Opportunistic Interference Alignment in MIMO Cognitive Networks. *IEEE Trans. Signal Process.*, 58(7):3728–3741, 2010.
[27] A. Poon, R. Brodersen, and D. Tse. Degrees of Freedom in Multiple Antenna Channels: a Signal Space Approach. *IEEE Trans. Inf. Theory*, 51(2):523–536, 2005.
[28] H. Rahul, H. Hassanieh, and D. Katabi. SourceSync: a Distributed Wireless Architecture for Exploiting Sender Diversity. In *ACM SIGCOMM*, 2010.
[29] A. Sayeed. Deconstructing Multiantenna Fading Channels. *IEEE Trans. Signal Process.*, 50(10):2563–2579, Oct. 2002.
[30] K. Tan, J. Fang, Y. Zhang, S. Chen, L. Shi, J. Zhang, and Y. Zhang. Fine-Grained Channel Access in Wireless LAN. In *ACM SIGCOMM*, 2010.
[31] K. Tan, H. Liu, J. Fang, W. Wang, J. Zhang, M. Chen, and G. M. Voelker. SAM: Enabling Practical Spatial Multiple Access in Wireless LAN. In *ACM MobiCom*, 2009.
[32] D. Tse and P. Vishwanath. *Fundamentals of Wireless Communications*. Cambridge University Press, 2005.