

# High-Rate WiFi Broadcasting in Crowded Scenarios via Lightweight Coordination of Multiple Access Points \*

Hang Qiu<sup>†</sup>, Konstantinos Psounis<sup>†</sup>, Giuseppe Caire<sup>‡</sup>, Keith M. Chugg<sup>†</sup>, Kaidong Wang<sup>†</sup>

<sup>†</sup>University of Southern California

<sup>‡</sup>Technische Universität Berlin

{hangqiu,kpsounis,chugg,kaidongw}@usc.edu<sup>†</sup>, {caire}@tu-berlin.de<sup>‡</sup>

## ABSTRACT

The enormous success of advanced wireless devices is pushing the demand for higher wireless data rates. The industry is satisfying this increasing demand by densely deploying large numbers of access points (APs). Unfortunately, unicast rates, especially in crowded scenarios, remain very low due to severe interference and time-sharing. However, one may take advantage of the broadcasting nature of wireless transmissions to offer high multicast rates. Motivated by this, we present coordinated broadcasting (Co-BCast), a system which coordinates multiple APs to provide participants of big events with high multicast rates that can support multiple high definition video streams.

Co-BCast requires no complicated time or frequency synchronization and no instantaneous channel state information. Yet, despite the time and frequency offsets among concurrent transmitters, the aggregate signal from the coordinated APs offers uniform coverage and high SINR to all users. We explore the challenges of such asynchronous transmissions through theoretical analysis and wireless experiments on software defined radio (SDR) testbeds. We use a number of PHY techniques to address those challenges and implement our coordination scheme on top of an 802.11 reference design. Both wireless experiments and large-scale simulations demonstrate that Co-BCast can achieve multicast rates in the order of 100Mbps even in the most crowded scenarios. These rates are orders of magnitude higher than the unicast and multicast rates achieved by uncoordinated transmissions.

## CCS Concepts

•Networks → Wireless local area networks;

## Keywords

Cooperative Transmission; Access Point Coordination; 802.11; WiFi Access in Crowded Scenarios

\*This work has been supported by NSF grant ECCS-1444060 and a Cisco Research Center grant.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

MobiHoc'16, July 04-08, 2016, Paderborn, Germany

© 2016 ACM. ISBN 978-1-4503-4184-4/16/07...\$15.00

DOI: <http://dx.doi.org/10.1145/2942358.2942372>

## 1. INTRODUCTION

The dramatic increase of the demand for wireless data [1] necessitates to use all possible approaches to satisfy this demand. Three main approaches are envisioned today: Moving to very high frequency bands where there is plenty of bandwidth, e.g. [26], freeing up additional bandwidth for wireless broadband access, e.g. [8], and, further increasing spectrum reuse via the densification of access points (APs) and base stations (BSs) allowing more concurrent transmissions on the same band.

We focus on the third approach which is already occurring in practice as is evident from the denser and denser deployments in enterprise WiFi settings, and from the direction towards micro-BSs that 5G standardization is heading. However, despite the use of smart power allocation, densification increases inter-cell interference which negates most gains from densification, as is most notably the case in stadiums and other crowded scenarios. Motivated by this, a number of recent works have proposed to coordinate nearby APs/BSs and employ advanced PHY layer techniques, e.g. MU-MIMO, to allow almost interference-free concurrent transmissions, see, for example, [11, 25, 32]. Unfortunately, all these works require a method to tightly synchronize the clocks of remote transmitters, such that time offsets (TO) and carrier frequency offsets (CFO) between concurrently transmitting nodes remain on check. This requirement makes them almost impractical in a WiFi setting.

Motivated by this, we ask the following question: Can we achieve high rates in a dense WiFi deployment scenario via concurrent transmissions *without* requiring any kind of clock synchronization? While this is not possible for unicast traffic, the broadcasting nature of wireless transmissions implies that we may be able to do so for multicast traffic. Our interest in this question is not merely academic but is motivated by multiple real-world scenarios where the majority of traffic is naturally of multicasting nature. Most telling, NFL has recently asked all NFL stadiums to deploy WiFi access such that the audience can watch replays of the current game as well as of other games taking place in other stadiums. In the two most recent Super Bowl games, about 1000 APs have been deployed within and around the stadium aiming to serve about 100,000 people. While this is clearly an extreme case, it is interesting to point out that the achieved unicast rates could barely sustain email communication and were too low to support even low definition video streaming using HTTP over TCP. It should also be mentioned that multicasting does not mean that *all* users must receive necessarily the *same* video stream. Well-known schemes such as Harmonic Broadcasting [18, 12] can be used to multiplex several different video streams into a single multicast transmission, such that any user can select one of the streams and start watching with minimal startup delay.

In this paper, we design a system which we call CO-BCAST that can deliver multicast rates in the order of 100Mbps even in the most crowded scenarios without any clock synchronization. A key feature of our system is that multiple APs can concurrently transmit the *same* packet while allowing all clients in the vicinity of those APs to successfully receive the packet despite the induced *cooperative* TOs (CTOs) and *cooperative* CFOs (CCFOs) between a receiver and these multiple transmitters<sup>1</sup> which render the wireless channel quite challenging.

We use software define radios (SDRs) to implement CO-BCAST. Specifically, we use the WiFi reference design of WARPv3 SDRs to implement an air trigger transmitted from an AP and used to initiate transmissions from multiple nearby APs within a cyclic prefix (CP). We also implement a coordination protocol which allows concurrent packet streaming among multiple APs and test the system using off the shelf WiFi clients.

We use a combination of theory, large-scale simulations and small-scale experiments using SDRs to study the performance of CO-BCAST. Specifically, we use SDR experiments to study the effect of various values of CTO and CCFO. CTO creates channel dips which cause symbol errors that standard convolutional codes used by today's WiFi chipsets cannot recover from. CCFO creates a time varying channel that eventually makes the channel estimation which occurs at the beginning of a packet grossly inaccurate. We find that employing modern low density parity check (LDPC) codes [27], like the ones described in the most recent WiFi standard but not implemented from any vendor today, can solve the problems caused by CTO. We also find that the packet transmission times and inter-symbol distances of some popular WiFi modes are such that receivers can deal with CCFO. To study the performance of CO-BCAST in large scale scenarios we resort to simulations where tens or hundreds of APs are deployed in conference halls, city centers, stadiums, etc. serving thousands of users. We find that in the presence of real world CTO and CCFO values, popular WiFi modes enhanced with LDPC codes can offer around 100Mbps data rates even in the most challenging environments.

The outline of the remainder of the paper is as follows. §2 introduces related work. §3 presents a motivating theoretical analysis of coordinated transmissions, while §4 conducts a series of wireless experiments on SDRs to explore and address fundamental PHY layer challenges due to CTO and CCFO. §5 presents a MAC design which enables a coordinated streaming service that is compatible with WiFi off-the-shelf devices. §6 evaluates the performance of CO-BCAST in large scale deployments and §7 draws the conclusion of the paper.

## 2. RELATED WORK

Wireless network access during crowded events has attracted a lot of research interest, ranging from traffic measurements [13, 29] to wireless video streaming [19, 17]. Indicatively, [13] analyzes both the WiFi and LTE traffic during crowded events and suggests a number of cross-layer optimization techniques. One way to ease the spectrum crunch is to broadcast the same unicast data of common interest in a multicasting fashion. Motivated by this, LTE eMBMS [21] tightly synchronizes the transmissions from multiple cells, both in time and frequency domain, to broadcast the same content through a single frequency network (SFN). Recently, Verizon had a preliminary proof of concept demo of eMBMS for video

<sup>1</sup>We use the terms cooperative TO and cooperative CFO to refer to the TOs and CFOs within concurrent transmitters to distinguish from the TO and CFO that always exists within a single transmitter and receiver, for which well known techniques exist to compensate.

service at NFL SuperBowl [5]. Unfortunately, in addition to the requirement for a centralized scheduler and shared clocking, it is very costly to deploy a dense network of micro LTE BSs for eMBMS purposes. In this paper, we explore the feasibility of such coordinated broadcasting using the already densely deployed WiFi APs without any centralized scheduler or shared clocking requirements.

In theory, the optimal solution under dense environments is to use cooperative systems such as distributed / coordinated MU-MIMO and massive MIMO systems, see, for example, [9, 10, 11, 25, 30]. These systems use channel state information (CSI) to precode and multiplex data streams to gain high throughput. In order for the symbols to be successfully combined, precise timing and frequency synchronization is needed. Unfortunately the overhead and cost of these systems are too large to be practical in the context of WiFi, due to not only the synchronization requirement, but also the requirement to constantly collect CSI especially as the number of users increases.

The idea to have multiple transmitters send the same bits is, of course, not new. The authors in [16] combine this idea with flooding and propose the concept of barrage relay networks in the context of MANETs with tactical communication applications. In [14] the authors introduce a novel flooding architecture for wireless sensor networks which exploits the constructive interference of 802.15.4 symbols for fast network flooding and implicit time synchronization. [31] explores how such a system scales as the number of relay hops increases. In [24] the authors study the alignment of OFDM symbols within the duration of a CP for cooperative transmissions purposes, [25, 11] use similar ideas for concurrent transmissions in the context of a distributed MU-MIMO system, and [20] enables basic coordination when using specific off-the-shelf WiFi chipsets. Last, [28] theoretically analyzes the effect of CTO and CCFO on top of OFDM and suggests scalable ways to achieve synchronization and calibration in large-scale networks in the context of OFDM based distributed MU-MIMO systems. While all of these prior works study concurrent transmissions and discuss CTO and CCFO effects, none explores in a systematic way the effect of CTO and CCFO for a wide range of values, and none does this over a WiFi reference design in a backward compatible way while achieving hundreds of Mbps without any clock synchronization.

## 3. UNCOORDINATED TRANSMISSIONS ANALYSIS

As a motivating example, we consider a simple analytically tractable scenario formed by the superposition at a given receiver of the same digitally modulated signal sent by two transmitters with a cooperative timing offset and frequency offset. Two main problems arise in the reception of the superimposed signals:

First, depending on the relative timing offset, signals may have similar power but opposite phase, thus canceling each other. Since the CTO and the received power depend on the propagation distance, and this ultimately depends on the relative position of the receiver with respect to the two transmitters, some locations are affected by signal mutual cancellation, causing “deep fades” of the useful received signal power.

Second, because of non-ideal frequency synchronization, there exists some CCFO that produces a time-varying phase offset in the received signal. Therefore, the effect of signal cancellation changes over time, producing time-varying fading even if the nodes are stationary. This creates an effective “Doppler spread” in the equivalent channel response as if the user was rapidly moving with respect to the two transmitters. In 802.11, the channel is estimated from the

pilot field at the beginning of each block, and it is assumed to be locally time-invariant across the block (in fact, 802.11 is a WLAN standard, designed to handle small mobility). Therefore, the channel time variations induced by the CCFO may cause significant performance degradation especially at the end of the blocks.

The passband modulated signal is given by

$$s(t) = \Re \left\{ x(t) e^{j(2\pi f_0 t + \phi_0)} \right\} \quad (1)$$

where  $x(t)$  is the (digitally modulated) complex envelope,  $f_0$  is the carrier frequency and  $\phi_0$  is the phase reference of the transmitter RF oscillator.

Consider two transmitters sending  $s(t)$  to a common receiver. Neglecting thermal noise (irrelevant to what we want to show), the relevant channel model is:

$$r(t) = a_1 \Re \left\{ x(t - \tau_1) e^{j(2\pi f_1 (t - \tau_1) + \phi_1)} \right\} + a_2 \Re \left\{ x(t - \tau_2) e^{j(2\pi f_2 (t - \tau_2) + \phi_2)} \right\} \quad (2)$$

where  $a_1$  and  $a_2$  are two real amplitude factors,  $\tau_1, \tau_2$  are the propagation delays due to different distances (in general) of the receiver from the two transmitters, and  $f_1, f_2$  are the two carrier frequencies. The carrier frequencies are in theory equal to a common frequency  $f_0$  but in practice they are slightly different because of implementation tolerances and non-ideal transmitter frequency synchronization, which is very difficult to achieve with 802.11 devices. Letting

$$A_i = a_i e^{-j(2\pi f_i \tau_i - \phi_i)}, \quad i = 1, 2$$

and assuming, without loss of generality, that the receiver demodulates with carrier frequency equal to  $f_1$ , simple algebra yields the complex envelope of the demodulated received signal in the form

$$y(t) = A_1 x(t - \tau_1) + A_2 e^{j2\pi\Delta f t} x(t - \tau_2), \quad (3)$$

where  $\Delta f = |f_2 - f_1|$  is the CCFO and  $|\tau_2 - \tau_1|$  is the CTO. This can be interpreted as the convolution of the transmitted complex envelope  $x(t)$  with the time-varying impulse response

$$h(t; \tau) = A_1 \delta(\tau - \tau_1) + A_2 e^{j2\pi\Delta f t} \delta(\tau - \tau_2). \quad (4)$$

The corresponding (time-varying) channel transfer function is obtained by taking the Fourier transform with respect to the delay variable  $\tau$ , and yields

$$H(t; f) = A_1 e^{-j2\pi\tau_1 f} + A_2 e^{j2\pi\Delta f t} e^{-j2\pi\tau_2 f}. \quad (5)$$

Notice that if  $\Delta f = 0$  this is the transfer function of an ordinary linear time-invariant channel. This means that  $H(t; f) = H(f)$  is independent of time  $t$ . Hence, this can be estimated on a block by block basis through the pilot preamble of 802.11, and used to perform frequency equalization in the usual way. However,  $H(f)$  may be strongly frequency selective. This depends on the relative amplitude of  $A_1$  and  $A_2$ , i.e., on the path coefficients  $a_1$  and  $a_2$ , which of course incorporate also (without loss of generality) the power imbalance of the transmitters.

Suppose now that  $\Delta f \neq 0$ . In this case, the transfer function  $H(t; f)$  depends on  $t$ . If  $1/\Delta f \approx T$  (or less than  $T$ ) where  $T$  denotes the slot duration, the change in time of the transfer function will be very significant over a block. This means that the multi-point transmission introduces both time-variations and frequency selectivity, and the 802.11 training and estimation scheme is likely to collapse, since it is not designed to cope with fast fading with significant variations over a block. If instead  $1/\Delta f \gg T$ , then the time variations are sufficiently slow such that the channel transfer function is approximately constant in time over one slot. In this

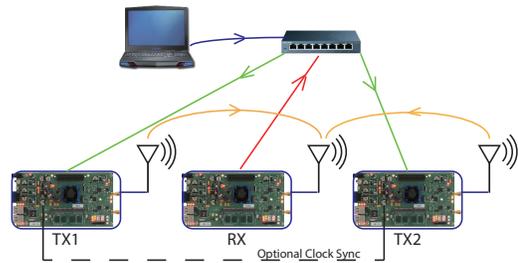


Figure 1—Experiment Setup.

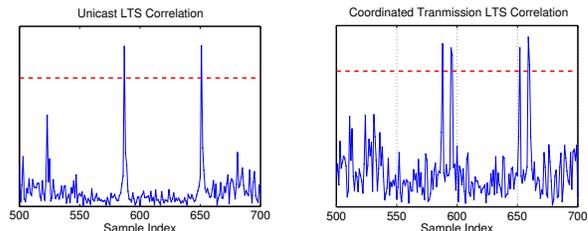


Figure 2—LTS Correlation for Block Boundary Detection.

case, the pilot estimation scheme of 802.11 yields accurate estimates across the whole block.

## 4. COOPERATIVE TRANSMISSION PHY

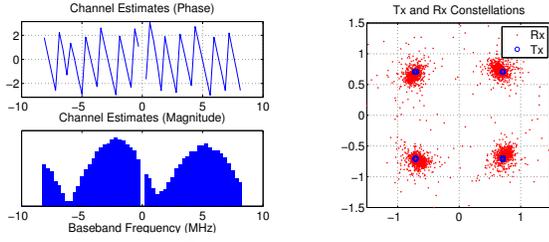
In addition to the simple theoretical model in §3, a series of wireless experiments are conducted to explore the performance and challenges in practice. Initially, we use a WiFi OFDM transceiver built from WARP SDR boards [6], to evaluate the coordinated transmission with various CTO and CCFO values. We first describe the experimental set-up, then demonstrate the PHY layer challenges arising from cooperative transmission, and finally demonstrate methods for mitigating degradations associated with these challenges.

### 4.1 Experimental Setup

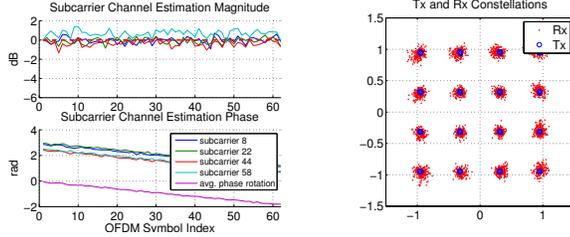
The system setup is presented in Figure 1. We use SDR FPGA boards as transmitters and receivers. Transceivers are controlled by a central server via Ethernet. Packets are sent from the server to each transmitter, transmitted and received through RF antennas from transmitter/s to receiver/s, and delivered back to the server for analysis. Each transceiver has the option to use its own clock or use the reference clock from the clock synchronization cable. While experiments are conducted with both same and different clocks, to have a controlled experimental environment and test different CTO and CCFO values in a consistent way, we do the following: We setup the transmitters to share the same reference clock for RF frequency and signal sampling, and different CTO and CCFO values are induced by the central server by adjusting the signal for each transmitter to emulate different offset values. Specifically, for CTO, a simple programmable delay is implemented by padding zeros before the short training symbols (STS). And, the CCFO is emulated by rotating each signal sample with a phase rotation increment for each sample equal to  $\Delta\theta = (\Delta f)2\pi/f_s$ , where  $f_s$  is the sample frequency and  $\Delta f$  is the CCFO in Equation (3).

### 4.2 Cooperative PHY Challenges

**Block Boundary Detection.** Block boundary detection occurs by correlating against known training probes to identify the start of the block. Typically, the largest amount of energy arrives at the shortest delay. However, with cooperative transmission we expect two large peaks corresponding to each transmitter which may cause degradation for traditional block boundary detection algorithms.



**Figure 3**—Channel Estimates and Constellation of Concurrent Transmission with CTO = 200 ns.



**Figure 4**—Sub-carrier Channel Estimates and Constellation of Unicast Transmission with CFO = 1000 Hz.

Specifically, for the 802.11 standard, the PHY header of a packet starts with 10 repetitions of STS and 2.5 repetitions of Long Training Symbols (LTS). The STS and LTS are used to correctly detect and locate the block boundary. Figure 2 shows the LTS correlation comparison between normal unicast packet and coordinated transmission with a cooperative time offset. In the case of unicast and a line of sight channel without any strong multipath component, the correlator output has two large correlation peaks corresponding to the LTS. In the case of Co-BCAST, the combination of signals yields four large correlation peaks corresponding to the LTS from both transmitters.

**Cooperative Time Offset (CTO).** To assess the impact of CTO, we first consider the case where CCFO = 0. For this case, the channel in Equation (5) is time-invariant. Furthermore, assuming that  $A_1 = A_2 = A$  for simplicity, the channel frequency response has magnitude

$$|H(f)| = A\sqrt{2(1 + \cos(2\pi f\Delta\tau))}. \quad (6)$$

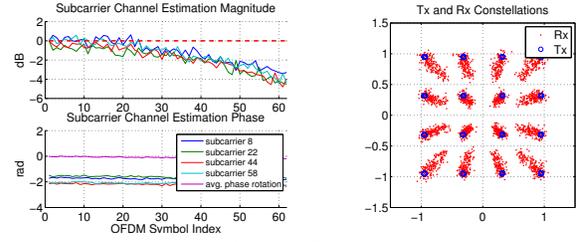
This illustrates the frequency-selective gain introduced by a nonzero CTO. Figure 3 shows the measured channel response and constellation of QPSK for concurrent transmission with CTO=200ns. The frequency-selective gain is compensated for in the receiver processing, so that the net effect on the constellation is larger noise variance (i.e., deviation from the reference point) for frequency bins with low gain. As predicted from the simple model in Equation (6) we observed experimentally that increasing the CTO increases the number of near nulls in the frequency domain.

**Cooperative Carrier Frequency Offset (CCFO).** For a single transmitter and receiver, standard carrier frequency offset (CFO) is compensated for with a frequency lock loop. For multiple transmitters, however, CCFO is not equivalent to this familiar CFO and a standard frequency tracker will not compensate for CCFO. For example, assuming that the CTO is zero, the model of Equation (4) simplifies to a time-varying complex gain

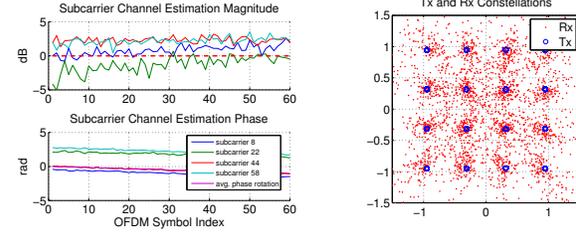
$$h(t) = A_1 + A_2 e^{j2\pi\Delta f t}. \quad (7)$$

Note that the magnitude of  $h(t)$  can vary significantly (e.g., for  $A_1 = A_2$  it can reach zero).

In a traditional WiFi system, a few sub-carriers are used to transmit known pilots for the receiver to track the CFO which manifests



**Figure 5**—Sub-carrier Channel Estimates and Constellation of Coordinated Transmission with CCFO = 1000 Hz.



**Figure 6**—Sub-carrier Channel Estimates and Constellation of Coordinated Transmission with CTO = 600 ns and CCFO = 1000 Hz.

itself as linear phase rotation in time. This is illustrated in Figure 4 where the channel amplitude and phase for several of these pilot sub-carriers is shown versus time. Because the amplitude is approximately constant with time and the phase is linear, the standard frequency tracking loop in the reference design tracks out the CFO and produces a clean constellation.

Figure 5 shows the corresponding plots for the case of two transmitters with CCFO and no CTO. In this case, we observed a near linear phase relation, but, as predicted by Equation (7), the amplitudes of the sub-carrier channels change significantly with time. The dashed line in Figure 5 is the initial value of the channel gains which are used by the receiver over the duration of the transmission. The net result of this condition is that the frequency tracking loop in the receiver tracks the linear phase, but the constellation is severely distorted for samples later in the transmission by this decreasing amplitude.

In practical cooperative scenarios, we expect both CTO and CCFO to exist. The effective channel, as modeled in Equation (5), will exhibit both frequency-selective gain (associated with nonzero CTO) and time variation (associated with nonzero CCFO). Again, our experimental data confirms this as shown in Figure 6 where the net effect on the received I/Q points may be viewed as a combination of the noise enhancement due to CTO (Figure 3) and the amplitude distortion due to CCFO (Figure 5).

### 4.3 Addressing Cooperative PHY Challenges

**Block Detection.** As discussed in §4.2, cooperative timing offset will introduce multiple large LTS correlation peaks. The 802.11 standard uses an 800 nsec CP so that the CTO in a benign channel should be accommodated. Our initial experiments showed catastrophic failure of the block boundary detection algorithm when two transmitters were used with CTO. This failure was due to cases where the algorithm detected the start of the block to be right after the second of the four peaks – i.e., it locked onto the first peak of the second incoming transmission. This will yield a non-causal channel estimate, which violates the principle of the CP. We altered the block boundary detection algorithm to search for and detect the correct peak. With this modification, substantially larger CTO values could be tolerated. The results in §4.4 and §5 utilize this modified block boundary detection algorithm.

**Forward Error Correction Coding (FEC).** WiFi utilizes convo-

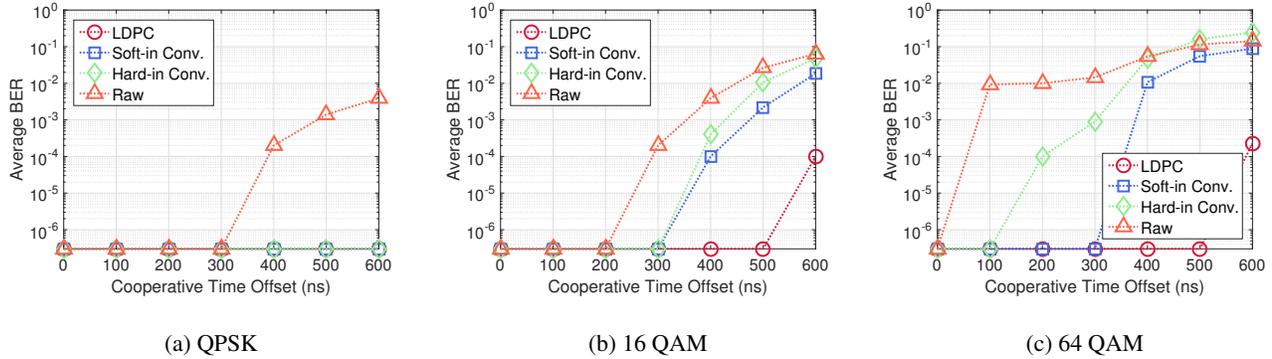


Figure 7—Avg. BER of Coordinated Transmission with Varying CTO.

lutional codes and has options for modern codes such as LDPC codes. Typically WiFi modems will use soft decision decoding of convolutional codes, but none of the existing real-time SDR WiFi designs [3, 4] provides soft decision decoding capabilities which requires implementing a soft constellation demapper. For convolutional codes, soft decision decoding typically provides 2.5-4 dB of additional receiver sensitivity depending on the modulation. Furthermore, LDPC decoders operate on the principle of iterative soft-in/soft-out decoding and therefore LDPC decoder typically requires a soft constellation demapper as well. Using non-realtime processing of experimentally collected data, we performed soft modulation demapping and decoding of convolutional [7] and LDPC codes (DVB-S2 codec [2]) to assess the impact of FEC on performance in the presence of cooperative transmissions.

**Phase Dithering.** The concept of phase dithering was introduced and analyzed in [22] for narrowband signals and is described in the context of a cooperative broadcast relay network in [16]. The idea is to combat the worst-case scenario where CTO yields a deep amplitude fade due to destructive combining. The phase dithering technique alters the transmitted phase in a pseudo-random manner to induce a time-varying amplitude which cycles through constructive and destructive combinations. A large modern code (e.g., an LDPC code) is then used to essentially obtain the performance of the average of these channel conditions. For the cooperative transmission case and due to the random CTO, the phase for each packet as perceived by the receiver is random, so packet-wise phase dithering is implemented automatically. Therefore, a code with block length containing many packets will implement the notion of phase dithering as described in [22].

#### 4.4 Experimental results

In this section, we explore the performance of coordinated transmission under various CTO and CCFO scenarios by examining the bit error rate (BER) of different coding schemes. Packet error rates are typically less than 5% and often less than 2%, which, assuming regular size packets and i.i.d. bit statistics, corresponds to about a  $10^{-6}$  BER. With this in mind, we repeat each experiment enough times such that if we get no bit errors in any of our samples we have at least 95% confidence that the BER is less than  $10^{-6}$ . In the following figures we plot these "zero" BER values with points a bit lower than the  $10^{-6}$  line for visual purposes.

**Cooperative Time Offset (CTO).** Using our experimental set-up, we set the CCFO to zero and vary the CTO from 0 to 600 ns. Since the 802.11 OFDM uses 16 samples of CP which is 800 ns, this CTO is within the modeled delay spread of the OFDM modem. However, Figure 3 (for varying values of CTO) and Equation (6) indi-

cate that deep spectral fades are induced by CTO, with the number of sub-carriers nearly nulled increasing with higher CTO values.

Figure 7 shows the average BER of QPSK, 16QAM and 64QAM coordinated transmissions with different CTO values using different coding schemes. As expected, increasing CTO generally degrades performance without coding. Using a convolutional code provides robustness up to the max value of 600 ns for QPSK, 400 ns for 16QAM and about 300 ns for 64 QAM. Utilizing an LDPC code allows for reliable reception up to 600 ns of CTO in all three cases. This can be interpreted as follows. The spectral nulls introduced by CTO will vary from packet to packet due to the inherent packet-wise phase dithering. The LDPC code length is many packets so it is able to extract performance approximately corresponding to the average over all channel gains. Since the convolutional code has a much smaller effective memory, it fails when too many deep fades occur within the decoding window.

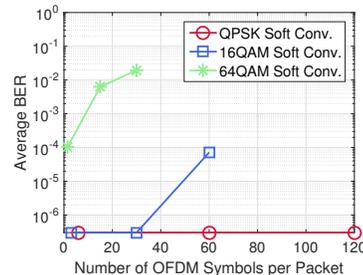


Figure 8—Avg. BER of Coordinated Transmission with Soft-in Convolutional Codes of Packet Size 144, 720, 1440 Bytes (CCFO=500Hz).

**Cooperative Carrier Frequency Offset (CCFO).** The range for CCFO to be expected from different AP transmissions depends on the oscillator specs of each AP. For the WARP boards used in our simulations, the worst case tolerances predict CCFO up to 2 kHz. However, when we measured the offset between two WARP boards, we observed an average of 500 Hz offset between the two oscillators.

As illustrated in Figure 5, CCFO causes a time-varying channel gain which, if estimated too infrequently, will cause severe performance degradation. Since the channel gains are estimated once per packet, this implies that larger transmission packet lengths will be more susceptible to CCFO. To elaborate on the effect of packet duration, we first study the performance for different packet sizes (number of OFDM symbols) for each modulation. For the reference of our specific experiments, one OFDM symbol consists of 64 regular samples and 16 samples of cyclic prefix. With a sample

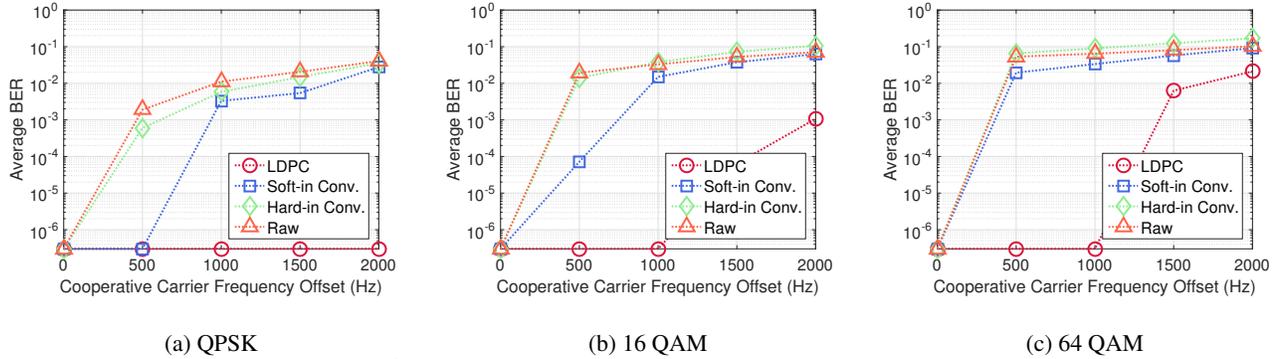


Figure 9—Avg. BER of Coordinated Transmission with Varying CCFO.

frequency of 20 MHz, the effective sample interval is 50 ns and one OFDM symbol duration is 4  $\mu$ s.

The maximum transmission unit (MTU) for 802.11 is 2304 bytes plus the packet header while typical data carrying packets are 1500 bytes. Without loss of generality, we transmit a packets of 144, 720 and 1440 bytes in QPSK, 16QAM, and 64QAM. Figure 8 shows that, as expected, a longer payload duration hurts performance. Also, despite the fact that larger constellations carry more bits and therefore require fewer symbol transmissions for a given packet length, they are still more sensitive to CCFO owing to their more dense symbol sets.

In the following, we illustrate how this trade-off affects the actual BER performance. We conduct a set of experiments with CTO being zero and CCFO varying from 0 to 2000 Hz. Note that we fix the regular packet size as 1440 bytes, but the packet duration (number of OFDM symbols transmitted) of each modulation is obviously different. Figure 9 shows the average BER of QPSK, 16QAM and 64QAM coordinated transmissions with different CCFO values. Generally, with higher CCFO and higher bit rate modulation, the average BER increases as well. While convolutional codes fail to provide robustness against CCFO, LDPC codes provide robustness for up to 2000Hz for QPSK, up to 1500Hz for 16QAM, and up to 1000Hz for 64QAM. For convolutional codes to provide robustness against such CCFO values, one approach would be to insert additional LTS for more frequent channel estimation.

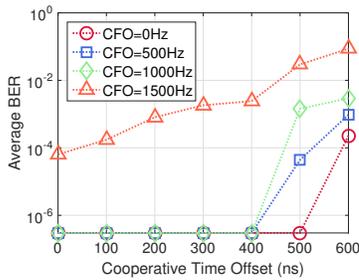


Figure 10—Avg. BER of 16QAM Coordinated Transmission with LDPC and Different CTO and CCFO.

In real WiFi scenarios, the combined signal of coordinated transmissions from asynchronous APs has both CTO and CCFO. To inspect the performance in real scenarios, we explore a wide range of CTO and CCFO combinations of different modulations in the worst case scenario where the two signals are received with the same power. Figure 10 shows the average BER of 16QAM with CTO varying from 0 to 600 ns and CCFO from 0 to 2000 Hz. The results suggest that, with a strong modern code, environments with

$CTO \lesssim 400$  ns and  $CCFO \lesssim 1$  KHz can be tolerated, and, when CCFO is  $\lesssim 500$  Hz, larger CTO values can also be tolerated. The next section implements CO-BCAST on a real-time 802.11 reference design. In these real time experiments the typical CTO is around 100 ns and the typical CCFO is around 500Hz, both of which are well within the operational ranges mentioned above.

## 5. MAC DESIGN AND IMPLEMENTATION

This section features the design and implementation of a real-time multi-AP coordination system. We build CO-BCAST on top of the WARP 802.11 reference design [3], which can perform regular functions as a WiFi AP to communicate with off-the-shelf WiFi devices. We implement within the MAC layer a simple over the air trigger to coordinate multi-AP transmissions, as well as a coordination protocol between APs to achieve real time CO-BCAST streaming.

### 5.1 Real-time Coordination Design

Advanced SDR platforms come with a full implementation of the whole WiFi stack. One of the fundamental challenges of such implementations is to meet the strict Inter-Frame Space (IFS) timing requirement of the 802.11 standard. This leads to a system-on-chip design that puts the entire PHY layer stack and MAC layer 802.11 Distributed Coordination Function (DCF) into an FPGA for fast computation and response, while implementing higher layer functionalities in software.

CO-BCAST coordinates multiple APs to broadcast the same packet concurrently. In order to completely remove OFDM inter-block interference, the arrival time of signals coming not only from different multiple paths between a transmitter and the receiver, but also of signals coming from other transmitters, should be within the WiFi OFDM CP of 800 ns. There are two sources of CTO. First, since transmitters will be, in general, at different distances from the receiver, the propagation delay of the line of sight signal from the various transmitters will vary by about 1 ns per 3m distance difference, or, by up to 100 ns considering the maximum range of WiFi. Another source of CTO is the free running clock and different hardware of each AP. In the rest of this section we describe the design and implementation choices that we made to ensure that the CTO is well within a CP.

**Over the air trigger.** Using an external trigger via the existing Ethernet cables is not a viable option as the processing time it takes for such a trigger to go from the network into the PHY layer is quite unpredictable. What is more, even if such a trigger arrives at the PHY layers of each AP simultaneously, an immediate transmission is not likely due to the asynchronous, distributed nature of the

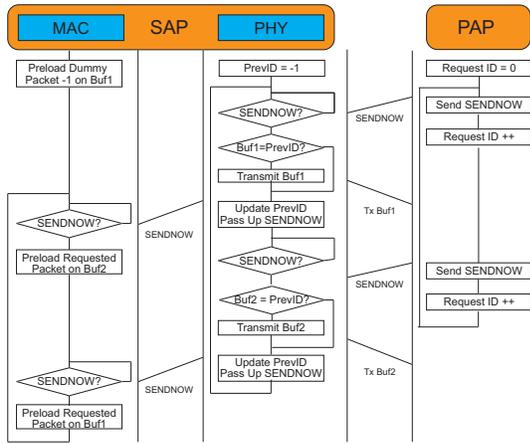


Figure 11—Timing Diagram of Streaming Service.

WiFi MAC protocol. Last, a centralized design that synchronizes APs and schedules their transmissions is clearly incompatible with WiFi.

Motivated by the above as well as recent works on AP coordination, see, for example, [24, 11], we utilize a trigger transmitted from one AP over the air. Specifically, considering a cluster of APs, one AP called the primary AP (PAP) transmits the air trigger, referred to as a SENDNOW message, to the rest of the APs called secondary APs (SAPs).

**Taking advantage of the ACK auto-responder.** According to the WiFi standard, the SAPs should respond with an ACK in  $16 \mu\text{s}$  (a SIFS guard slot). We utilize this ACK auto-responder feature of the standard to have all SAPs respond to the SENDNOW message by broadcasting the same data packet.<sup>2</sup> A request sequence number is appended to the SENDNOW message to specify which packet all SAPs should transmit to ensure the same data are transmitted from all SAPs.

Since SAPs have to respond after an SIFS, the SENDNOW message has to be recognized as early as possible. To ensure this, we use the first two bits of the MAC header to flag such a message. Currently, all 802.11 traffic has these bits set to 00. By raising the first bit and setting the destination address as the broadcast address, the SENDNOW message is essentially a broadcasted MAC header with the first two bits set to 10.

**Ping-pong buffering for data packet preloading.** ACKs are short known sequences of bits that can be generated on the fly and meet the SIFS deadline. Full data packets are not and it would take too long to search a packet and write it to the transmission buffer in time. For this reason, data packets have to be pre-loaded into the transmission buffer, which we achieve by using ping-pong buffers as described below.

Consider two ping-pong buffers,  $\text{Buf}_1$  and  $\text{Buf}_2$ . Figure 11 shows a timing diagram of the interaction between the PAP and the SAPs and the use of the ping pong buffers. The main idea of the preloading process is for the SENDNOW message to indicate the ID of the next packet to be transmitted, leaving enough time to pre-load the packet while the current packet is being transmitted. Specifically, at initialization all APs fill up  $\text{Buf}_1$  with dummy data which they release after receiving the first air trigger,  $\text{SENDNOW}_0$ , indicating the ID of the first actual packet to be transmitted. While the

<sup>2</sup>It is easy to have the PAP as well to participate in the coordinated transmission following the SENDNOW message but we don't implement this functionality for simplicity.

dummy data are released, the SAPs pre-load the first actual packet to  $\text{Buf}_2$ . Next,  $\text{SENDNOW}_1$  will trigger the transmission of  $\text{Buf}_2$  while the requested packet from  $\text{SENDNOW}_1$  is prepared in  $\text{Buf}_1$  and so on as so forth.

There is a chance that some APs have the requested packet ready to be transmitted in one of the ping-pong buffers whereas other APs don't. When a SAP does not have the right packet ready, it will simply defer from transmitting during this coordinated transmission.

Note that to form a streaming service, such as live video feeds in an event stadium, data files are distributed through Ethernet to a backlog queue at every AP. As long as there is a backlogged packet to be broadcasted, the PAP constantly contends for the channel and upon winning the channel it transmits the next SENDNOW message.<sup>3</sup> In our testbed a video server is constantly feeding the APs with packets via their Ethernet connection. Last, when the backlog queue becomes full due to bursts, jitter, etc. we evict older packets based on a FIFO principle, and, when a particular sequence number is currently served, packets with smaller sequence numbers are evicted as well.

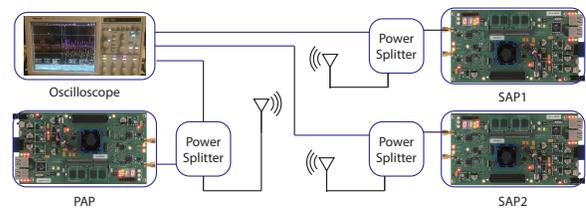


Figure 12—Real-Time Measurement Setup.

## 5.2 Experimental Results in WiFi

Figure 12 shows the basic setup of our experiments. In order to verify the correct operation of the system and deduce real world numbers for the CTO, we use an SMA cable and connect a power splitter to divide the signal into two identical copies, one for the RF antenna, and one for an oscilloscope. We use a high precision oscilloscope (DPO71254B from Tektronix) with maximum analog bandwidth 12.5 GHz and sampling rate 100GS/s.

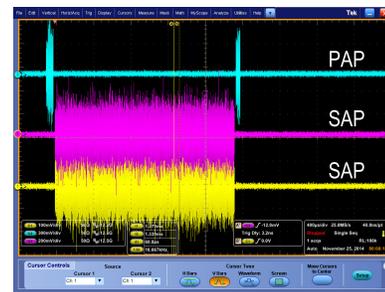


Figure 13—Waveforms of PAP and SAPs output during a real time WiFi coordinated transmission as seen on an oscilloscope.

**Coordination Scheme.** To test if a packet can be correctly received without error in CO-BCAST and access whether the CTO of our coordination scheme is small enough for the signal to be decoded, we perform a simple experiment with one PAP sending the SENDNOW messages as well as serving as the receiver of the data

<sup>3</sup>Clearly, the PAP contends for the channel with uplink traffic and, perhaps, other APs that do not belong to the PAP's cluster. To guarantee contention fairness for the broadcast traffic, the PAP can adjust its contention window in order to increase the chance of grabbing the channel, see, for example, [20] where the authors have implemented this feature in the context of a real world WiFi chipset.

packets, and two SAPs participating in the coordinated broadcast. Figure 13 shows the waveform observed from the oscilloscope. The two SAPs send out the data packet after receiving the PAP’s air trigger and the PAP responds with an ACK (we use the PAPs MAC address as the destination address in this experiment), confirming that the packet was received correctly. Note that for the signals to be combined and decoded correctly, the MAC headers from different transmitters have to be exactly the same as well. We ensure this is the case by using the same default broadcasting address as the source address for both SAPs.

To acquire real world data about the resulting CTO, we perform more than 14,000 coordinated transmissions and report the cumulative distribution function (CDF) of the CTO in Figure 14. From this data it turns out CO-BCAST has a mean CTO of 97.32 ns while the maximum is 473.24 ns, way below the OFDM CP duration of 800 ns. While these numbers are hardware specific, we conjecture that the WiFi reference design in an SDR testbed is less optimized than a commercial WiFi chipset and thus the CTO will most likely be smaller in a commercial setting than what we report. Specifically, the CTO results from the fact that the radio turnaround times (from receive to transmit mode) may not be exactly the same in different boards, the WiFi stack implementation may introduce different delays depending on the state of each board when the trigger is received, the clocks of each board may have different accuracy, all of which make the actual time between the reception of the trigger till the initiation of the transmission to be a bit off the 16  $\mu$ s that the standard prescribes. (Note that a 100ns time offset constitutes less than 1% discrepancy on the actual duration of the SIFS interval.)

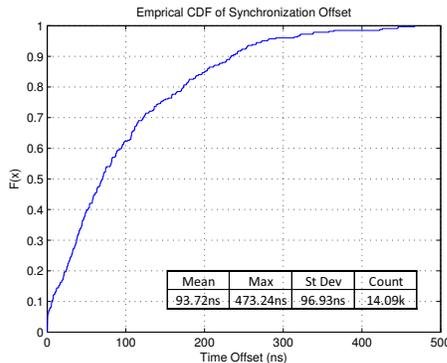
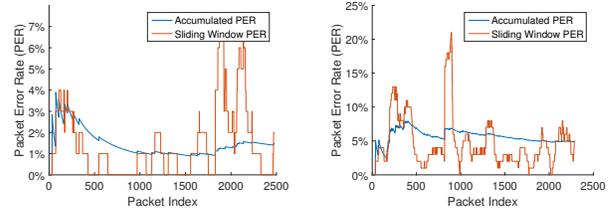


Figure 14—CDF of Coordinated Transmission CTO.

**Streaming packets.** To evaluate the entire interaction protocol integrating the PHY, MAC, and network / application layers, we build a streaming service to pair with coordinated transmissions. Similar to the setup in Figure 1, we use a laptop as a streaming server, constantly broadcasting sequenced packets to each AP through wired Ethernet. For the sake of simplicity, all APs store broadcast packets in one specific queue for broadcasting. The PAP constantly contends for the channel and triggers the SAPs for coordinated transmission. As before, we use the PAP as the receiver to monitor the streaming performance. (In a commercial implementation the PAP would join the coordination transmission after sending the trigger and the receiver/s would be WiFi clients.)

Figure 15 shows the percentile of successfully received packets as a function of the packet sequence number. We report numbers from two setups, one inside an anechoic chamber, and the other from a corridor of an office building, where the SAPs are transmitting in QPSK each using their own free running oscillator/clock. Note that since the WiFi reference design of the WARP SDR only implements hard-in convolutional codes [3], the probability of er-



(a) Anechoic Chamber

(b) Corridor

Figure 15—Packet Error Rate of QPSK Streaming with Hard-in Convolutional Codes in Anechoic Chamber and Corridor.

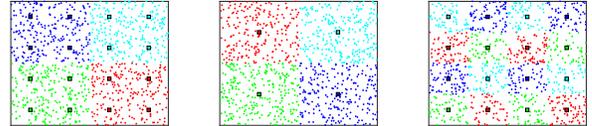


Figure 16—CO-BCAST, RELOCATED PAP, BCAST Channel Allocation in Conference Hall.

ror is higher than it would be with a modern LDPC-like or even a soft-in convolutional code implementation.

## 6. LARGE SCALE SIMULATIONS

We focus on dense environments where a large number of users is served by a large number of APs, like is the case in conference and concert halls, airports, university campuses, city centers, and stadiums. In these cases, if the density of APs is low, each AP will associate with a large number of users resulting in low rates due to time sharing. In order to improve performance, APs are usually densely deployed so that each of them serves less users. Special care is required when assigning channels to nearby APs to reduce as much as possible inter-cell interference.

In such crowded scenarios it is often the case that users are interested on the same media content, e.g. stadium audiences watching game replays. CO-BCAST is designed to offer to such audiences multimedia multicasting services.

To investigate the performance of CO-BCAST under real world large scale environments we consider two scenarios: (i) a densely populated conference hall of dimension 30m x 30m with 300 active users, and (ii) a stadium of 150m diameter with 10000 active users, where by active users we are referring to users that are actively connected on the WiFi network. In each scenario, users and APs are distributed uniformly in space (in the second scenario restricted in the audience seats area) and channels are chosen to minimize inter-cell co-channel interference, see Figure 16 and Figure 19 where in the case of CO-BCAST APs are grouped into clusters and all APs within a cluster are assigned the same channel. The rate of each individual user is computed based on the individual Signal to Interference and Noise Ratio (SINR) values derived from the transmission power, the distance between user and surrounding APs, the path-loss exponent, and typical fading and shadowing. Specifically, we assume a transmission power of 90 dB above noise floor and the widely accepted WINNER-II pathloss model [23]. Note that in the case of CO-BCAST all coordinated APs within a cluster contribute to the receiving power of all associated users.

We compare CO-BCAST against three alternative multicasting/broadcasting schemes. First, we consider uncoordinated broadcasting, where each AP broadcasts to its clients within its channel. We refer to this scheme as BCAST. Given that we are interested in

APs	Avg. Rate (Mbps)	Users / AP	AP Distance (m)
4	1.918925	75	15
16	2.158524	19	7.5
36	2.55452	8	5
64	2.505313	5	3.75

Figure 17—Unicast Performance in Conference Hall.

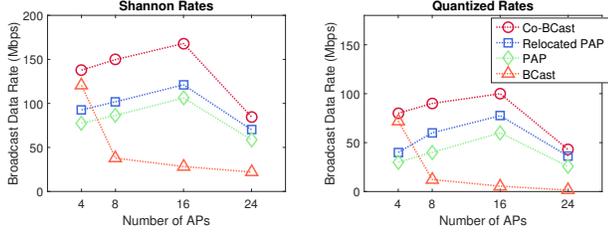


Figure 18—Shannon and Quantized Rates in Conference Hall.

broadcasting rates and time sharing is not an issue, the large number of APs deployed to support decent unicast rates creates unnecessary interference. To reduce interference, one may shut down a number of APs letting only a portion of them to broadcast. Specifically, assuming APs belong to clusters for the purposes of CO-BCAST, we consider a scenario where all SAPs are shut down and only PAPs broadcast. We call this scheme PAP. Note that in practice re-associating the clients of SAPs to PAPs is very costly and not even possible when both unicast and broadcast traffic is of interest, but we include this scheme nevertheless for comparison purposes. We further optimize this scheme by relocating PAPs in the middle of the clusters, clearly not an option in the real world, to boost the performance of alternative approaches as much as possible. We refer to this scheme as RELOCATED PAP. In all four schemes, the broadcasting rate of each AP/cluster is dictated by the worst SINR among all associated users.

Without loss of generality, we assume four 20MHz non-interfering WiFi channel in all scenarios. In addition, we presented not only Shannon rates but also quantized rates based on the 802.11ac Modulation and Coding options as shown in Table 1 [15].

Table 1—802.11ac Modulation and Coding Pairs with Minimum SINR Requirement.

802.11ac MCS Index	Modulation	Code Rate	SINR
0	BPSK	1/2	$\geq 2$ dB
1	QPSK	1/2	$\geq 5$ dB
2	QPSK	3/4	$\geq 8$ dB
3	16-QAM	1/2	$\geq 12$ dB
4	16-QAM	3/4	$\geq 15$ dB
5	64-QAM	2/3	$\geq 18$ dB
6	64-QAM	3/4	$\geq 21$ dB
7	64-QAM	5/6	$\geq 24$ dB
8	256-QAM	3/4	$\geq 27$ dB

**Conference Hall.** We first compute the unicast rates for various AP densities. Figure 17 shows that as the number of APs increases and thus the number of associated users per AP decreases, the average user rate increases due to more airtime per user. However, as the number of APs continues to increase, the shorter inter-AP distance creates significant inter-cell interference, and the average user rate drops accordingly.

Figure 18 shows both the Shannon rates and quantized rates performance in the conference hall scenario with a cluster size of 4. For quantized rates we use the SINR thresholds dictated by the 802.11 standard, see Table 1. For uncoordinated broadcasting (BCAST), the rate drops as the number of AP increases due to co-channel interference. On the other hand, clustering/coordinating APs or shutting off some of the APs effectively reduces interference such

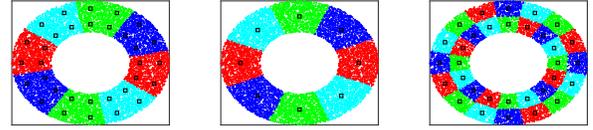


Figure 19—CO-BCAST, RELOCATED PAP, BCAST Channel Allocation in Stadium.

APS	Su(Mbps)	users per AP	AP distance(m)
8	0.033848	1250	53.033009
16	0.071042	625	37.5
32	0.1273	313	26.516504
64	0.20014	156	18.75
128	0.311114	78	13.258252
256	0.495758	39	9.375
512	0.795542	20	6.629126
1024	1.1793	10	4.6875

Figure 20—Unicast performance in stadium.

that the average broadcasting throughput keeps increasing when the number of APs is below 16 (CO-BCAST, PAP, RELOCATED PAP). CO-BCAST, as expected, offers the highest rates. Specifically, CO-BCAST has an average (maximum) gain of 3.7x (5.9x) over BCAST, of 1.6x (1.8x) over PAP, and of 1.5x (1.7x) over RELOCATED PAP.

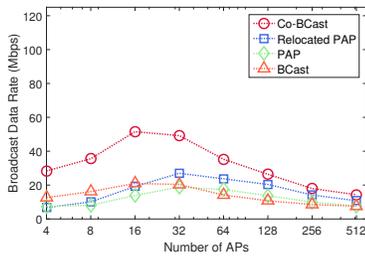
**Stadium.** With a stadium topology (Figure 19), the average unicast rates per user increase as the number of APs increase (Figure 20). Figure 21 shows the broadcast data rate of CO-BCAST, RELOCATED PAP, PAP and BCAST with a cluster size of 4, 16, and 36. CO-BCAST achieves its maximum rates when the number of APs equals the cluster size times the number of channels, since this ensures no co-channel interference. CO-BCAST clearly achieves the highest rates. For example, it achieves a maximum gain of 26.5x over BCAST and 7.4x over PAP at cluster size 16.

## 7. CONCLUSIONS

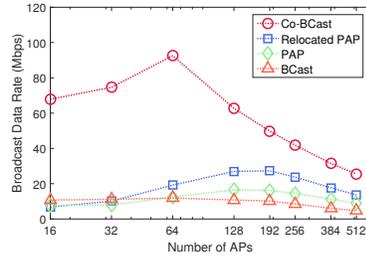
In this paper we establish that nearby APs can be architected to broadcast in a coordinated fashion achieving multicasting rates of 100Mbps even in the most challenging scenarios like that of a large crowded stadium. These rates would allow tens of HD video streams to be multicasted. No clock synchronization is needed to achieve those rates but merely an over-the-air trigger signal coupled with a lightweight coordination protocol and modern LDPC-like codes to deal with the induced cooperative time and carrier frequency offsets.

## 8. REFERENCES

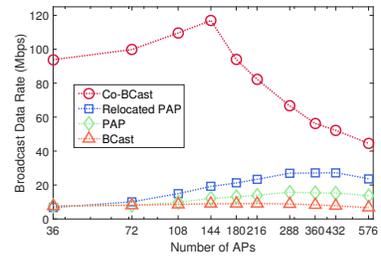
- [1] Cisco global mobile data traffic forecast. [http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white\\_paper\\_c11-520862.html](http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html).
- [2] LDPC codec in DVB S2. <http://www.mathworks.com/help/comm/examples/dvb-s-2-link-including-ldpc-coding.html>.
- [3] Mango communications 802.11 reference design. <http://mangocomm.com/802.11/>.
- [4] USRP and GNURadio. <http://www.ettus.com/>.
- [5] Verizon eMBMS Trial at SuperBowl 2014. <https://techzine.alcatel-lucent.com/playing-crowd-verizon-lte-multicast-and-superbowl-2014>.
- [6] WARP Project. <http://warpproject.org>.
- [7] IEEE 802.11: Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Std 802.11-2007*, pages 1–1076, June 2007.



(a) AP cluster size: 4



(b) AP cluster size: 16



(c) AP cluster size: 36

**Figure 21**—Shannon Rates in Stadium with Different Cluster Size

- [8] Report and order and second further notice of proposed rulemaking,. Federal Communications Commission 15-47 GN Docket No. 12-354, 2015.
- [9] E. Aryafar, N. Anand, T. Salonidis, and E. W. Knightly. Design and experimental evaluation of multi-user beamforming in wireless lans. In *Proc. of MobiCom '10*.
- [10] H. V. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire. Achieving high data rates in a distributed mimo system. In *Proc. of Mobicom '12*.
- [11] H. V. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire. Airsync: Enabling distributed multiuser mimo with full spatial multiplexing. *IEEE/ACM Trans. Netw.*, 21(6):1681–1695, 2013.
- [12] L. Engebretsen and M. Sudan. Harmonic broadcasting is bandwidth-optimal assuming constant bit rate. In *Proc. 13-th annual ACM-SIAM Symposium on Discrete Algorithms (SODA '02)*. Citeseer, 2002.
- [13] J. Erman and K. Ramakrishnan. Understanding the super-sized traffic of the super bowl. In *Proc. of IMC '13*.
- [14] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with glossy. *IPSN '11*, pages 73–84. IEEE, 2011.
- [15] M. Gast. *802.11Ac: A Survival Guide*. O'Reilly Media, Inc., 1st edition, 2013.
- [16] T. R. Halford and K. M. Chugg. Barrage relay networks. In *Information Theory and Applications Workshop (ITA), 2010*.
- [17] I.-H. Hou and P.-C. Hsieh. Qoe-optimal scheduling for on-demand video streams over unreliable wireless networks. In *Proc. of MobiHoc '15*.
- [18] L.-S. Juhn and L.-M. Tseng. Harmonic broadcasting for video-on-demand service. *Broadcasting, IEEE Transactions on*, 43(3):268–271, 1997.
- [19] L. Keller, A. Le, B. Cici, H. Seferoglu, C. Fragouli, and A. Markopoulou. Microcast: Cooperative video streaming on smartphones. In *Proc. of MobiSys '12*.
- [20] S. Kumar, D. Cifuentes, S. Gollakota, and D. Katabi. Bringing cross-layer MIMO to today's wireless LANs. In *Proc. of SIGCOMM '13*.
- [21] D. Lecompte and F. Gabin. Evolved multimedia broadcast/multicast service (embms) in lte-advanced: overview and rel-11 enhancements. *Comm. Magazine, IEEE*, 50(11):68–74, 2012.
- [22] D. K. Lee and K. M. Chugg. A pragmatic approach to cooperative communication. In *In Proc. of MILCOM '06*.
- [23] J. Meinilä, P. Kyösti, T. Jämsä, and L. Hentilä. Winner ii channel models. *Radio Technologies and Concepts for IMT-Advanced*, pages 39–92, 2009.
- [24] H. Rahul, H. Hassanieh, and D. Katabi. Sourcesync: A distributed wireless architecture for exploiting sender diversity. In *Proc. of SIGCOMM '10*.
- [25] H. Rahul, S. Kumar, and D. Katabi. Megamimo: Scaling wireless capacity with user demands. In *Proc. of SIGCOMM '12*.
- [26] S. Rangan, T. S. Rappaport, and E. Erkip. Millimeter-wave cellular wireless networks: Potentials and challenges. *Proc. of the IEEE*, 102(3):366–385, 2014.
- [27] T. J. Richardson and R. L. Urbanke. Efficient encoding of low-density parity-check codes. *Information Theory, IEEE Transactions on*, 47(2):638–656, 2001.
- [28] R. Rogalin, O. Y. Bursalioglu, H. Papadopoulos, G. Caire, A. F. Molisch, A. Michaloliakos, V. Balan, and K. Psounis. Scalable synchronization and reciprocity calibration for distributed multiuser mimo. *IEEE Trans. on Wireless Communications*, 13(4):1815–1831, 2014.
- [29] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, S. Venkataraman, and J. Wang. A first look at cellular network performance during crowded events. In *Proc. of SIGMETRICS '13*.
- [30] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong. Argos: Practical many-antenna base stations. In *Proc. of Mobicom '12*.
- [31] Y. Wang, Y. He, X. Mao, Y. Liu, and X.-Y. Li. Exploiting constructive interference for scalable flooding in wireless networks. *IEEE/ACM Trans. Netw.*, 21(6):1880–1889, Dec. 2013.
- [32] S. Yun, L. Qiu, and A. Bhartiya. Multi-point to multi-point mimo in wireless lans. In *Proc. of INFOCOM '13*.