Experimental Demonstration of Nullforming From a Fully Wireless Distributed Array

Ben Peiffer[1], Raghu Mudumbai[1], Sairam Goguri[1], Soura Dasgupta[1,2], Anton Kruger[1]
[1]Dept. of Electrical and Computer Engineering, University of Iowa, Iowa City, IA 52242
{benjamin-peiffer, raghuraman-mudumbai, sairam-goguri, anton-kruger, soura-dasgupta}@uiowa.edu
[2]Shandong Computer Science Center, Shandong Provincial Key Laboratory of Computer Networks

***Abstract*—We consider distributed nullforming using an array of wireless transmitters that coordinate their transmissions to achieve destructive interference at a designated receiver. We describe the *first* experimental demonstration of distributed nullforming to a target receiver from an array of three distributed transmitters using (mostly) off-the-shelf hardware and simple and standard signal processing techniques. We are motivated by the goal of using distributed arrays to achieve increased spectrum reuse through interference cancellation. Our results show interference suppression in excess of 25dB over uncoordinated transmission. We build on our recent experimental demonstration of beamforming from a distributed antenna array after estimating and compensate for the combined effects of unknown propagation channels, hardware mismatches and clock drifts between the array nodes. The transmitters do not share clocks or have any wired back channels. They coordinate achieve nullforming entirely using in-band wireless message exchanges. Thus these methods can be implemented on portable mobile devices rather than being limited to Base Stations.**

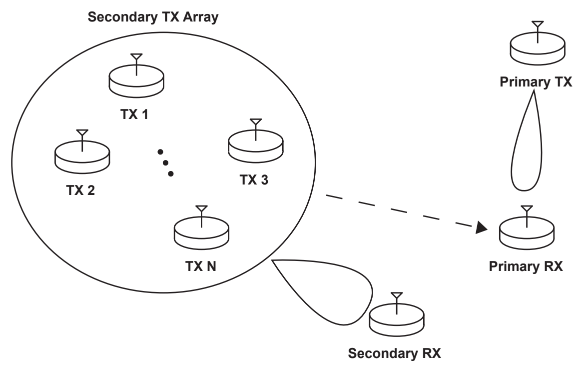

Fig. 1. Interference cancellation using distributed nullforming for a cognitive radio network.

I. Introduction

We report *the first* experimental demonstration of nullforming from a fully wireless distributed array of transmitters to a designated receiver. Nullforming [1] refers to a class of techniques where multiple antennae coordinate their transmissions to achieve destructive interference at designated receivers. As we explain in the sequel, nullforming is very difficult to realize in a distributed wireless setting. Yet it can be a powerful enabling technology for many network capabilities. Thus, an array can transmit jamming signals to disable hostile devices while protecting friendly devices by forming nulls to them. More generally, nullforming can cancel interference at specific locations allowing the increased spatial reuse of spectrum. Thus, in a cognitive radio network [2], a secondary user can use nullforming to reuse the spectrum of a primary user. Fig. 1 illustrates such a network, where a distributed array acting as a secondary transmitter communicates with a secondary receiver while forming a null at the primary receiver.

A. Background and Related Work

Distributed arrays, comprising a network of cooperating wireless transceivers emulating a virtual multi-antenna device, have recently attracted a lot of interest in industry [3] and academia [4] as they open up the possibility of using powerful MIMO techniques on a large scale without being limited by the size and form factors. Many synchronization techniques have been developed to compensate unknown propagation channels and oscillator offsets and to calibrate the array for retrodirective transmission [5].

The simplest application for a distributed array is beamforming [6] where the array transmissions combine coherently at a desired receiver; the directivity of the resulting transmission yields SNR gains that scale with the number of elements in the array. Many distributed beamforming algorithms have been developed based on explicit channel feedback [7], aggregate 1-bit feedback [8], or reciprocity [5]. Several experimental demonstrations of these algorithms exist, [7], [9].

More sophisticated array processing methods such as spatial multiplexing for distributed arrays involving transmission of multiple independent signals to one or more receivers have also been studied [10], [11]. In contrast to beamforming, which is known to be robust to moderately large synchronization errors [6], the more sophisticated methods such as nullforming and spatial multiplexing require very precise cancellation of interfering signals [1]. The resulting stringent synchronization requirements for the array nodes has presented a formidable challenge to a practical implementation of these methods. Experimental demonstrations of spatial multiplexing for WiFi [12], [13], and similar claims for LTE cellular networks [14]

This work is in part supported by US NSF grants EPS-1101284, ECCS-1150801, CNS-1329657 and CCF-1302456, ONR grant N00014-13-1-0202 and under the Thousand Talents Program of the State and Shandong Province, administered by the Shandong Academy of Sciences, China.

have also been reported; however, these all involve arrays of wired Base Station nodes and make heavy use of high bandwidth backhaul links for coordinating the array transmissions.

B. Key Challenges and Contribution

The lack of demonstration of wireless distributed null, as opposed to beamforming has several reasons. In a distributed setting each node has its own oscillator. Even if pre-synchronized, these oscillators quickly drift out of synchrony, [20]- [23]. As demonstrated analytically in [1], nullforming requires levels of synchronization that are *orders of magnitude more stringent beamforming.* For example a reasonable beam can be sustained with upto 30^o phase disparity. Even a much smaller level of phase offset will be fatal to nullforming. To an extent this can be attributed to the fact that signal cancellation is much less robust than reinforcement. Further, beams can be attained simply by phase alignment. Nulls require intricate combinations of both phases and magnitudes.

Our demonstration directly implements the iterative nullforming algorithm of [1]. It uses the hardware and software used in our previous experimental demonstration of beamforming [9], which have been refined and optimized to meet the more stringent demands of the nullforming. While we show the practical feasibility of distributed nullforming and open the way for powerful new capabilities and applications, we emphasize, that these are preliminary results. The removal of the assumptions and limitations of our implementation are an active topic of ongoing work.

Section II has the system architecture and the theory behind our nullforming algorithm. Section III details the testbed used for our implementation, and experimental results illustrating the nullforming process. Section IV concludes by identifying open problems for future work using the example of the cognitive radio application of Fig. 1.

II. System Architecture

Our experimental setup consists of three cooperating transmitter nodes nullforming to a single receiver. The transmitters use the iterative algorithm of [1], where each node transmits at full power and makes periodic phase adjustments guided by feedback from the receiver to which they seek to form a null. Similar to our previous beamforming experiment [9], each message in our implementation consists of fixed-length QPSK modulated packets and each packet contains a known preamble sequence that is used for packet detection and timing synchronization. The transmitters and receiver periodically exchange messages according to the time-slotted TDMA schedule as shown in Fig. 2, which is very similar to the ones used in [9]:

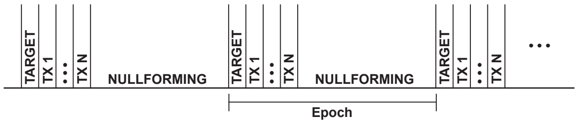

Fig. 2. Time-slotting schedule for a general nullforming system. Each epoch consists of a brief period of time for feedback and training, and a much longer period of time in which the null is applied.

(1)Feedback Message. At the beginning of each epoch, the receiver broadcasts a message containing its estimate of channel state information (CSI) of the transmitters from previous epochs and also a small amount of aggregate feedback consisting of a single complex number needed to drive the nullforming process. The transmit nodes use the preamble of this message to establish timing and carrier frequency synchronization, and also obtain CSI estimates from the payload.

(2) Training Messages. Each transmitter takes turns to transmit a message to the receiver. The receiver uses these messages to estimate CSI for each transmitter and averages the estimates over time using an Extended Kalman Filter (EKF).

(3) Nullforming. In the rest of the epoch, the array nodes transmit a common message with a complex gain adjustment based on the received feedback. This transmission will form a null at the desired receiver.

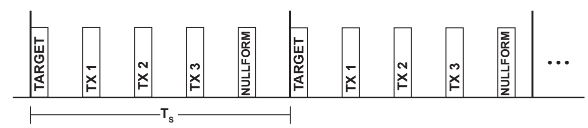

Fig. 3. Time-slotting schedule for the experiment with three transmitters and one target. In this experiment, the epoch length, T_s, was 100 ms.

Note that the feedback and training message exchanges in steps ((1), (2)) represent overhead in the synchronization and nullforming process, and in a practical application, we would like to maximize the duration of the nullforming step (3) when the array actively transmits while placing a null at the receiver. Usually the maximum duration of the epoch is limited by the stability of the oscillators used by the array transmitters, which determines how frequently the array nodes must be resynchronized to maintain the stable phase relationship required for nullforming. For the purposes of our experimental demonstration, each epoch consists of one feedback packet from the receiver, three separate training packets (one from each of the three array nodes), and a single nullforming packet. This is illustrated in Fig. 3. Thus our demonstration is not designed to be very efficient in terms of overhead and is intended simply to illustrate the nullforming process.

A. Iterative Nullforming Algorithm

In the k'th epoch, each transmitter $i \in \{1, 2, 3\}$ applies the complex weight $x_i[k] = Ae^{j\theta_i[k]}$ to its outgoing baseband message signal. Note that the magnitude A of the weight $x_i[k]$ remains constant over time i.e., the array nodes adjust only the phase of their transmissions to achieve the desired null. Specifically, the nodes iteratively update their phase as follows:

$$\theta_i[k+1] = \theta_i[k] - \mu r_i \mathrm{Im}\Big[e^{-j(\theta_i[k]+\phi_i[k])} s[k]\Big] \qquad (1)$$

where the complex channel gain of transmitter i to the receiver is $h_i \equiv r_i e^{j\phi_i}$, and $s[k]$ is a complex number representing the magnitude and phase at the receiver of the nullforming packet transmitted together by the array nodes in the previous epoch. Note that in order to implement (1), transmitter i needs knowledge of its own channel state i.e., r_i, ϕ_i, as well as the common feedback $s[k]$, and it obtains all of these quantities from the feedback message from the receiver.

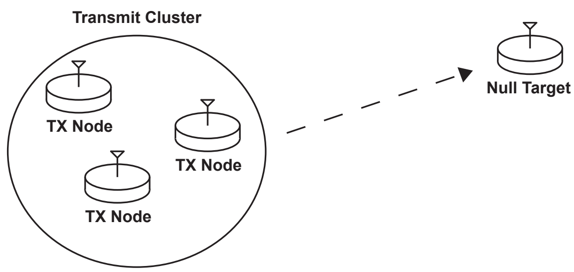

Fig. 4. The experimental setup consists of three transmitters that form a null to one receiver.

The algorithm (1) is known to converge to a null and to be very robust to errors in estimated channel state information [1]; specifically, estimates of the channel magnitude and phase responses $r_i,\ \phi_i$ can be off by up to 6 dB and $90°$ respectively without any performance loss in terms of the achievable null depth. In our experiment, all transmitters have identical hardware, quasi-isotropic antennas and are separated by comparable distances from the receiver. This allows us to simply assume the r_i's to be equal and absorb them into the μ parameter. The "step-size" parameter μ in (1) controls the rate of convergence of the algorithm. If μ is chosen too large, the algorithm will not converge to a null, but generally the algorithm works well over a wide range of μ values.

The algorithm is, very sensitive to time varying phase errors and thus requires precise frequency synchronization between the transmitters. For this, each transmitter uses the known preamble in the receiver feedback packet to make an estimate of its relative frequency offset and phase offset with the receiver in each epoch. These frequency and phase estimates are smoothed by filtering them using an EKF similar to the one in [15], and the resulting estimates are used to derive a frequency correction term that is then applied to all outgoing transmissions. This correction, makes each transmitter's packets arrive at the receiver with very close to zero frequency offset.

B. Challenges of Nullforming

As noted earlier, our nullforming implementation uses many of the building blocks from our previous beamforming implementation in [9]. We now highlight a couple of subtle challenges that we encountered with the nullforming problem that required significant improvements over [9], and that were not considered in the theoretical analysis in [1].

1) Detecting and decoding nullforming messages: The algorithm (1) requires a reliable estimate of $s[k]$ at each transmitter, and the availability of such an estimate is assumed in [1]. The complex number $s[k]$ represents the magnitude and phase at the receiver of the nullforming packet transmitted together by the array nodes, and is estimated by the receiver and sent as part of its feedback message in each epoch. In practice, as the nullforming process proceeds, by definition, the nullforming message signal gets weaker and weaker, and as a result, it becomes harder for the receiver to detect these messages and measure their strength. The nullforming algorithm (1) is thus self-limiting in a fundamental sense.

In our implementation, all nodes detect incoming transmissions using a simple threshold test on the output of a matched filter correlating against a known preamble sequence. The choice of a threshold for this test involves making a trade-off between missed packets (false negatives) and spurious packet detections (false positives). In practice, this threshold needs to be chosen close to the noise floor to avoid performance issues arising from excessive spurious packet detections.

Once the nullforming signal strength drops below this threshold, in our implementation, the receiver can no longer reliably detect the nullforming packets, and therefore is unable to make new estimates of $s[k]$. The transmitters are then unable to make any further adaptations to their phases, until channel variations and clock drifts degrade the quality of the null at which point the receiver is again able to detect the nullforming messages and resume the phase update process.

Thus, the nullforming process in our implementation reaches a steady-state where the signal power level fluctuates around the detection threshold. Careful choice of the thresholds in our experiment allowed us to drive the null signal power to around -25 dB compared to the incoherent power level (i.e., the average power level corresponding to uncoordinated array nodes transmitting with random phases).

2) Precise timing synchronization.: Consider a distributed array that cooperatively transmits a message signal consisting of QPSK modulated symbols as in our experiment. The transmissions of the array nodes need to be timed precisely so that each node is transmitting the same modulated symbol at any given time instant. In practice, small timing errors between nodes manifest themselves as inter-symbol interference (ISI) e.g. one array node may start sending symbol $n+1$ while the other nodes are still sending symbol n.

If the timing errors are much smaller than the symbol duration, the performance loss because of ISI can usually be tolerated and the messages can be reliably decoded. This was the case in our beamforming experiment in [9] where reliable decoding and large beamforming gains could be achieved even with timing errors on the order of a tenth of a symbol duration.

However, with nullforming, even a small amount of ISI is intolerable because it represents an uncancelled interference signal at the null target. In our implementation, we used a recently developed joint delay-Doppler estimation algorithm [16] that achieves timing accuracy close to the Cramer-Rao lower bound to essentially eliminate timing errors.

III. Experimental Results

The basic setup for this experiment is shown in Fig. 4. The hardware is based on the USRP N2X0 software defined radio platform with the WBX RF daughterboard [17] at a center frequency of 800 MHz. All nodes used computers running Linux software and the open-source GNU Radio platform

[18]. The only custom hardware is an external oscillator board with an oven-controlled oscillator (OCXO) from Abracon LLC [19]. We first present results from a fully wireless experiment for nullforming and then, for comparison, results from an experiment where all nodes share a single OCXO clock signal.

A. *Fully Wireless Transmit Nullforming*

Fig. 5 depicts a sequence of packets representing two epochs observed at the receiver after the algorithm has reached steady state. Each epoch has five packets: the first is the receiver's own transmission of the feedback message observed over the isolation of its antenna switch. The next three have training messages from each transmitter. The fifth packet is the joint transmission from the array.

Fig. 6 shows signal strengths of the transmitters and the nullforming signal in each epoch over a 35 second experiment run.It also shown "incoherent power level" inferred from the signal strengths of the transmitters. The nullforming algorithm converges to an amplitude $20-25$ dB lower as compared to the incoherent power level in this experiment

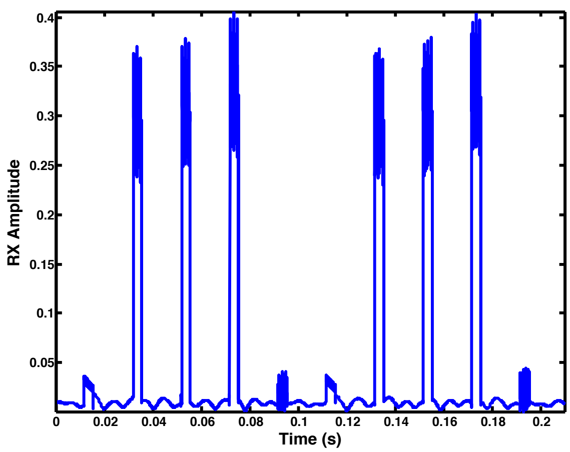

Fig. 5. A sequence of packets over two epochs as seen at the receiver. The y-axis is the A/D converter scale with a maximum value of 1.

.

B. *Transmit Nullforming with Common Local Oscillator*

In the experiments with common OCXO, the LO frequency offset between nodes are zero by definition; thus a comparison with the experiment without common clocks will allow us to see the effect of imperfect frequency offset correction. It turns out that this effect is significant. Fig. 7 shows the evolution of the nullforming algorithm over one 25 second experiment run. The algorithm in this experiment reaches steady state within 5 seconds and maintains a null that is consistently $25-35$ dB below the incoherent power level, which improves on the independent clocks experiment by $5-10$ dB.

IV. Conclusion

We have described a series of experiments that successfully demonstrates nullforming from a fully wireless distributed array. Our current implementation needs to be developed

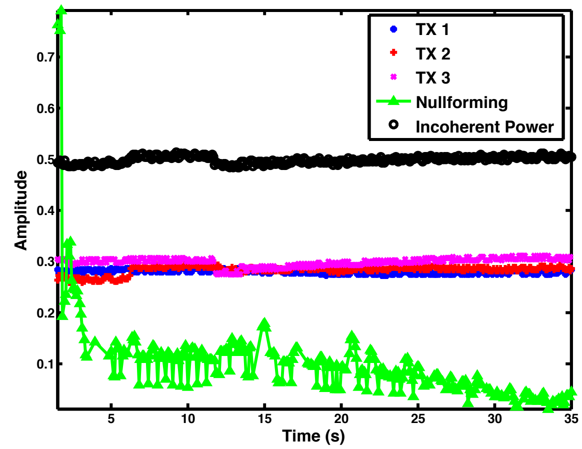

Fig. 6. A plot of the average nullforming amplitude of each packet compared to the incoherent amplitude and the amplitudes of the individual transmitters in the most recently received packets. The y-axis is the A/D converter scale with a maximum value of 1.

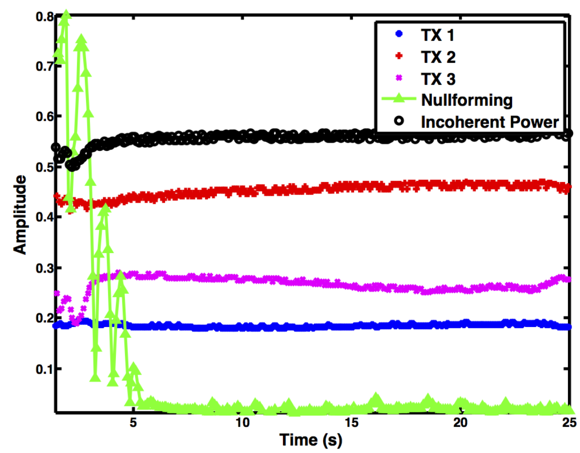

Fig. 7. A plot analogous to the one shown in Fig. 6 for an experiment with common clocks.

further before it can be used in a practical application such as the cognitive radio network of Fig. 1. Two such limitations in particular are worth highlighting as important challenges that remain to be addressed in future work. First, in our implementation, even after the nullforming algorithm has reached steady-state, the training signal transmission in each epoch still cause interference at the null target, which is the primary receiver in the cognitive radio network. This is undesirable; for a well-designed cognitive radio system, the secondary user should be completely invisible to the primary user. The second limitation is the fact that our implementation requires active and ongoing cooperation from the primary receiver in the form of feedback. Retrodirective methods where the nodes of a distributed array obtain channel estimates to an uncooperative receiver simply by opportunistically observing its transmissions offer one possible way to overcome the above limitations and there has been interesting recent work in this area [5]. Designing nullforming algorithms based on retrodirective methods is an important topic for further study.

REFERENCES

[1] A. Kumar, R. Mudumbai, S. Dasgupta, M. M. U. Rahman, D. R. B. III, U. Madhow, and T. P. Bidigare, "A scalable feedback mechanism for distributed nullforming with phase-only adaptation," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 1, pp. 58–70, March 2015.

[2] K. B. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 878–893, May 2009.

[3] R. Irmer, H. Droste, P. Marsch, M. Grieger, G. Fettweis, S. Brueck, H. P. Mayer, L. Thiele, and V. Jungnickel, "Coordinated multipoint: Concepts, performance, and field trial results," *IEEE Communications Magazine*, vol. 49, no. 2, pp. 102–111, February 2011.

[4] A. Ozgur, O. Leveque, and D. N. C. Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Transactions on Information Theory*, vol. 53, no. 10, pp. 3549–3572, Oct 2007.

[5] P. Bidigare, D. R. Brown III, U. Madhow, R. Mudumbai, A. Kumar, B. Peiffer, and S. Dasgupta, "Wideband distributed transmit beamforming using reciprocity with endogenous relative calibration," in *Asilomar Conference on Signals, Systems, and Computers, 2015 IEEE*. IEEE, 2015.

[6] R. Mudumbai, G. Barriac, and U. Madhow, "On the feasibility of distributed beamforming in wireless networks," *Wireless Communications, IEEE Transactions on*, vol. 6, no. 5, pp. 1754–1763, 2007.

[7] P. Bidigare, M. Oyarzyn, D. Raeman, D. Chang, D. Cousins, R. O'Donnell, C. Obranovich, and D. Brown, "Implementation and demonstration of receiver-coordinated distributed transmit beamforming across an ad-hoc radio network," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, Nov 2012, pp. 222–226.

[8] R. Mudumbai, B. Wild, U. Madhow, and K. Ramchandran, "Distributed beamforming using 1 bit feedback: from concept to realization," in *Proceedings of the 44th Allerton conference on communication, control and computation*, 2006, pp. 1020–1027.

[9] B. Peiffer, R. Mudumbai, A. Kruger, A. Kumar, and S. Dasgupta, "Experimental demonstration of a distributed antenna array pre-synchronized for retrodirective transmission," in *Information Sciences and Systems (CISS), 2016 50th Annual Conference on*, March 2016, pp. 1–6.

[10] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser mimo channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 461–471, Feb 2004.

[11] A. Kumar, R. Mudumbai, and S. Dasgupta, "Scalable algorithms for joint beam and null-forming using distributed antenna arrays," in *Global Communications Conference (GLOBECOM), 2014 IEEE*. IEEE, 2014, pp. 4042–4047.

[12] H. Rahul, S. S. Kumar, and D. Katabi, "Megamimo: Scaling wireless capacity with user demand," *Proc. of ACM SIGCOMM 2012*, 2012.

[13] H. V. Balan, R. Rogalin, A. Michaloliakos, K. Psounis, and G. Caire, "Airsync: Enabling distributed multiuser mimo with full spatial multiplexing," *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 6, pp. 1681–1695, 2013.

[14] A. Forenza, S. Perlman, F. Saibi, M. Di Dio, R. van der Laan, and G. Caire, "Achieving large multiplexing gain in distributed antenna systems via cooperation with pcell technology," 2015.

[15] F. Quitin, M. M. U. Rahman, R. Mudumbai, and U. Madhow, "A scalable architecture for distributed transmit beamforming with commodity radios: Design and proof of concept," *Wireless Communications, IEEE Transactions on*, vol. 12, no. 3, pp. 1418–1428, 2013.

[16] S. Goguri, R. Mudumbai, B. Peiffer, S. Dasgupta, and U. Madhow, "Near optimal algorithm for joint delay and doppler shift estimation in the presence of colored noise," in preparation.

[17] "Ettus research," 2015, [online] http://www.ettus.com.

[18] "Gnu radio," 2015, [online] http://www.gnuradio.org.

[19] "Oven controlled crystal oscillator datasheet," 2014, [online] http://www.abracon.com/Precisiontiming/AOCJY4.pdf.

[20] L. Galleani, "A tutorial on the two-state model of the atomic clock noise," *Metrologia*, vol. 45, pp. S175S182, December 2008.

[21] D. R. Brown, R. Wang and S. Dasgupta, "Channel State Tracking for Large-Scale Distributed MIMO Communication Systems", *IEEE Transactions on Signal Processing*, pp. 2559 - 2571, 2015.

[22] F. Quitin, M. M. Rahman, R. Mudumbai and U. Madhow, "A Scalable Architecture for Distributed Transmit Beamforming with Commodity Radios: Design and Proof of Concept" *IEEE Transactions on Wireless Communications*, pp. 1418 - 1428, 2013.

[23] D. R. Brown, R. Mudumbai and S. Dasgupta, "Fundamental limits on phase and frequency tracking and estimation in drifting oscillators", in *Proceedings of ICASSP*, pp. 5225-5228, March 2012, Kyoto, Japan.