Multiple Access with Asynchronous Broadcasting in Wireless Cooperative Networks

Antonios Argyriou*†
*Department of Computer and Communication Engineering, University of Thessaly, Volos, 38221, Greece
†The Center for Research and Technology Hellas (CERTH), Thessaloniki, 57001, Greece

***Abstract*—In this paper we present a multiple access scheme for wireless networks that allows multiple nodes to broadcast their packets simultaneously. The simultaneously broadcasted packets are superimposed with a random pattern at each network node that receives them. A subset of the network nodes act as cooperative relays and forward the locally superimposed packets without decoding them. Each destination node recovers the desired packet with a successive interference cancelation (SIC) and decoding algorithm that exploits the pattern of the superimposed transmissions. Different flavors of the basic scheme are investigated and combine amplify-and-forward (AF) from the relays, and also retransmissions from the initial sources.**

I. Introduction

When wireless signal transmissions interact unintentionally at a particular receiver, then this phenomenon significantly affects communication performance. The undesired transmission is well known as interference [1]. Interference is a fundamental problem in wireless networks and as a result, it has been investigated in the literature from several angles [1]. More recently there is a trend to perceive potentially interfering transmissions across competing network nodes as a tool to increase the performance of the wireless network as a whole. Physical layer network coding (PLNC) is one technique [2], [3], [4]. PLNC is based on the intentional creation of interference through targeted transmissions of specific packets. This means that a wireless node transmits a packet in such a way that it interferes with another primary transmission, when it knows that the transmitted packet can be removed at the node that needs the primary packet. However, PLNC is still an opportunistic mechanism for exploiting interference that requires very specific network topologies and traffic flows [2], [3]. There is a need to exploit interference in in a more generic setting. In this paper we investigate a more general approach for utilizing interfering transmissions for the benefit of the complete network and this is accomplished by revisiting the channel access scheme.

A distributed wireless network typically features multiple nodes that access the communication channel through a contention mechanism. This particular class of multiple access (MA) schemes works very well in practice but it does not exploit to the fullest one basic property of wireless networks, that is their broadcast nature. One work that considered these issues was the NDMA protocol that was presented by Tsatsanis *et al.* [5]. NDMA is a reactive protocol since it is only activated when a collision takes place. With this protocol the source nodes involved in the collision retransmit their initial packets so that decoding is possible at the destinations. In recent extensions of the previous work, Lin and Petropulu in [6] proposed NDMA/ALLIANCES that adds the element of cooperation since relay nodes may forward the collided packet for recovering from a collision. Nevertheless, in both these works the authors assume first that the packets are colliding perfectly without any time arrival differences between them, and second that both time and cooperative diversity are exploited only when a collision happens and not otherwise.

Antonios Argyriou would like to acknowledge the support from the European Commission through the Marie Curie Intra-European Fellowship WINIE-273041 and the STREP project CONECT (FP7ICT257616).

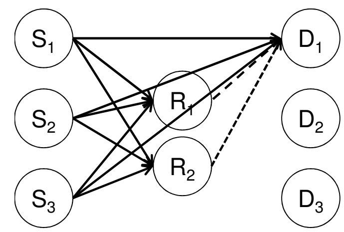


Fig. 1. Example network model with three simultaneously transmitting sources and two relays. Each source has one designated destination. Solid lines indicate transmissions during the broadcast phases. Dashed lines indicate transmissions in different forwarding phases.

In this paper our goal is to exploit collisions but in such a way that they occur as part of the normal system operation. This can be accomplished by a new multiple access scheme for wireless networks. To achieve the above, we develop first a multi-source cooperative multiple access scheme and second a supportive successive interference cancelation (SIC) and decoding algorithm that allow us to support collisions in a more general setting: (1) Multiple sources with independent packet flows are supported, (2) sources can transmit packets of an arbitrary length, (3) packet transmissions are asynchronous.

II. System Model

Before we proceed with the description of the system model, we define several important terms that are used throughout this document. To avoid the use of the negative term collision, at this point we introduce the term *over-the-air superimposed transmissions* (OST). Furthermore, we also define the *symbol slot* as the basic time unit that we consider in this paper and it corresponds to the transmission time of a physical layer (PHY) symbol. A *transmission phase* is a system defined parameter that denotes the time period where a node (source or relay) can transmit a packet and it consists of many symbol slots. This is essentially similar to a time-division multiple access (TDMA) slot but in our system multiple sources can transmit during this phase. If the length of a packet from source n is L_n in symbols, and the duration of the transmission phase is T then it means that this node has $K_n = T - L_n$ empty symbol slots during that transmission phase. During a transmission phase a packet is allowed to start the transmission at any time instant as long as it complies with the length of the transmission phase. If the number of sources that take part in a transmission phase is N then it must be $L_n \leq T,\ \forall n \in N$. For the rest of this paper we examine the events in one *transmission round*, i.e. a time duration where a specific group of nodes desire to transmit a packet and it consists of several transmission phases that we describe in detail in the next paragraph. Regarding the network topology, an example can be seen in Fig. 1. Our system model does not require fixed roles for the network nodes that can be sources, relays, or destinations. The separation of these three groups of nodes is only done for notational convenience. The group of sources is denoted by $\mathcal{N}$, the group of destinations by $\mathcal{D}$, and finally the group of used relays is $\mathcal{M}$.

III. The ACMA Protocol

The *asynchronous cooperative multiple access (ACMA)* scheme exploits both spatial and time diversity of the superimposed broadcasted packet transmissions. The main idea of this protocol is that all sources broadcast during one transmission phase and in the next transmission phase one or more relays forward the received signals. Fig. 2 presents an example for the protocol behavior in the time domain. In this example there are two broadcast phases from all the N=3 sources and one forwarding phase from a single relay. In the lower part of the figure the superimposed signal that is received at a single destination and the relay is shown. During the forwarding phase, the relay broadcasts the locally received superimposed signals after it applies the appropriate power scaling. The minimum number of required relaying (M) plus broadcast (B) phases is equal to the number of sources that transmit concurrently in order to allow for linear decoding complexity.

During the broadcast phase the sources are allowed to transmit by being synchronized only at the symbol level. Thus, the transmitted symbols are superimposed in the time domain in an arbitrary fashion. To capture this behavior we follow a different approach for describing the transmitted packet. Let $x_n[t]$ be the t-th complex modulated symbol that source n transmits as part of a packet. Also $s_n[t]$ denotes the signal that is transmitted from the source n during the symbol slot t and it can either contain the actual modulated symbol or simply nothing. Therefore, for a source n it will be:

$$s_n[t] = \begin{cases} x_n[t] & \forall t \in \mathcal{L}_n \\ 0 & \forall t \in \mathcal{K}_n \end{cases}$$

The group of symbols $\mathcal{L}_n$ corresponds to the L_n symbols of the packet that were transmitted while $\mathcal{K}_n$ denotes the group of empty symbol slots.

Now let us proceed with the detailed description of the cooperative protocol with the help of mathematical notation. The channel transfer functions between source n, relay m, and destination d are denoted as $h_{n,m}$, $h_{m,d}$, and $h_{n,d}$. Recall that the group of all sources that transmit concurrently is $\mathcal{N}$ and this is used in our derivations next. The received signal at a relay m during symbol slot t, and only for the broadcast transmission phase B_j, is

$$y^{(B_j)}_{\mathcal{N},m}[t] = \sqrt{P} \sum_{n \in \mathcal{N}} h_{n,m} s_n[t] + w^{(B_j)}_m[t], \tag{1}$$

where $\sqrt{P}$ is the amplitude of the transmitted symbol at each source, and $w^{(B_j)}_m[t]$ denotes the additive white Gaussian noise (AWGN) at the relay m. Similarly, the received signal at destination d during the same broadcast transmission phase B_j is the linear combination of the transmitted symbols from all the sources and can be written as

$$y^{(B_j)}_{\mathcal{N},d}[t] = \sqrt{P} \sum_{n \in \mathcal{N}} h_{n,d} s_n[t] + w^{(B_j)}_d[t]. \tag{2}$$

For one broadcast phase there are multiple forwarding phases depending on number of relays M. In each of the forwarding phases a relay m broadcasts the received signals given in (1) by applying a power amplification factor g_m so as to maintain the power constraint [7]. The received signal at d for forwarding phase F_j can now be written as

$$y^{(F_j)}_{m,d}[t] \quad = \quad g_m[t] h_{m,d} y^{(B_j)}_{\mathcal{N},m}[t] + w^{(F_j)}_d[t]. \tag{3}$$

In the above expression we see that the relay amplifies and forwards the broadcasted signal that was received from all the sources in a previous phase. Note also that (3) corresponds to the received signal from one relay. Regarding the power amplification coefficients they are defined as

$$g_m[t] = \sqrt{\frac{P}{P \sum_{n=1}^{N} \gamma_{n,m}[t] + \sigma^2}}, \tag{4}$$

where $\gamma_{n,m} = |h_{n,m}|^2$. We define also the vector $\mathbf{g}$ for the power gains of all the used relays as

$$\mathbf{g} = \begin{bmatrix} g_1 & \dots & g_m & \dots & g_M \end{bmatrix}.$$

For writing the input-output equations of the channel during each phase, we also need to define the column vector that describes the transmitted signal

$$\mathbf{s}[t] \quad = \quad \begin{bmatrix} s_1[t] & s_2[t] & \dots & s_T[t] \end{bmatrix}^T,$$

and the channel gain from all sources to destination d as the vector

$$\mathbf{h}_{\mathcal{N},d} = \begin{bmatrix} h_{1,d} & \dots & h_{N,d} \end{bmatrix}^T.$$

Now we are ready to write in matrix form the received signals we just described for destination d. We have that during a broadcast phase B_j the received signal is:

$$y^{(B_j)}_{\mathcal{N},d}[t] = \sqrt{P} \cdot \mathbf{h}_{\mathcal{N},d} \cdot \mathbf{s}[t] + w^{(B_j)}_d[t] \tag{5}$$

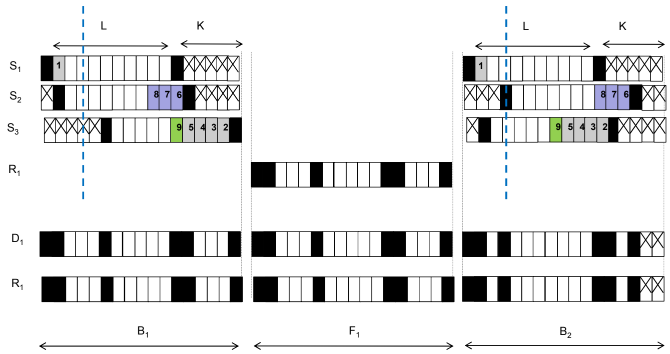


Fig. 2. Example of packet transmissions from two sources and one relay with ACMA. The dark-shaded blocks indicate the symbols that belong to the preambles and postambles of a packet. The blocks with an internal X indicate the empty symbol slots. The vertical dashed lines depict the fourth symbol slot where misalignment can occur but only in different broadcast phases.

For the forwarding phase F_j the signal is described as

$$y_{m,d}^{(F_j)}[t] = \sqrt{P} \cdot \mathbf{g} \cdot \mathbf{h}_{m,d} \cdot \mathbf{s}[t] + g_m h_{m,d} w_m + w_d[t], \quad (6)$$

where in this case it is

$$\mathbf{h}_{m,d} = \left[\begin{array}{ccc} h_{1,m}h_{m,d} & .. & h_{N,m}h_{m,d} \end{array} \right]^T .$$

Note that the above signals correspond to a specific symbol slot but the expressions are the same for all $t \in \mathcal{T}$.

IV. Two-Dimensional SIC Across Broadcasted Superimposed Transmissions

The decoding algorithm we describe now was designed specifically so that both the assumptions of perfect synchronization across the broadcasting phases and that of fixed packet lengths are eliminated. The reason is that there is no guarantee that the alignment of the same packets will be the same in different broadcast phases. For example in Fig. 2 the dashed vertical lines show how different symbols, that belong to the same packets, are aligned with a different pattern in the two different broadcast phases namely B_1 and B_2. The only assumption for the proposed algorithm is that nodes maintain symbol-level synchronization, i.e. there is alignment across symbol level boundaries. This later assumption is not limiting since it is a prerequisite for point-to-point communication and has also been used by recent works on asynchronous cooperative systems [8].

A. X-Dimension SIC Algorithm

The decoding algorithm described in this section is different from the classic notion of SIC since it cancels completely decoded symbols even before a mechanism like e.g. ML detection is used. The main idea of the algorithm is that cancelation is applied across multiple and inconsistently aligned symbol streams along the X dimension. Fig. 2 is used for explaining how the algorithm works. If we look the second transmitted symbol from S_1, it is marked as 1 and is "clear" during phase B_1 since it is only superimposed with a known preamble from source S_2. Therefore, this symbol can be detected on its own and it is the first to be detected. However, it cannot aid in the detection of another symbol during phase B_1. Therefore, the signal received during B_1 is parsed *backwards* from right to left starting from the postamble. In this way symbols marked as 2,3,4,5 can be detected immediately because they are not superimposed with anything else. This information is stored and the algorithm proceeds to the next broadcast phase, i.e. B_2. The same process starts which allows the detection of symbols marked as 6 and 7. However, at this point we can also detect symbol marked 8 since we can cancel the symbol marked as 2. Once the algorithm cannot detect any more symbols in this way it starts by parsing again the first phase B_1. With similar logic the known symbol 6 now allows the decoding of the symbol marked as 9. When during the complete forward and backward parsing of the symbol slots of each broadcast phase does not lead to any more symbol decoding the algorithm stops. At this point the algorithm has collected at most N equations with maximum N unknowns for a specific symbol slot t while the algorithm has canceled along the X dimension all the previously detected symbols and are contained in the matrix of estimated symbols $\hat{\mathbf{x}}$. Also the helper matrix $\tilde{\mathbf{x}}$ contains the unknown variables that remain to be detected after this process is finished, and is passed to the Y-dimension SIC decoding algorithm. If we generalize our observations from the previous description, we see that the core function of this algorithm is that it cancels complete linear equations.

B. Y-Dimension SIC Decoding Algorithm

We now describe the detection algorithm executed at the destination node for each symbol slot t. In Fig. 2 we see that the algorithm detects symbols aligned along the Y dimension, an observation that provided it its name. It is important to stress that after the X-Dimension SIC has been applied we use the notation $\tilde{\mathbf{y}}[t]$ to denote the transmitted signal after the canceled symbols have been removed and with $\tilde{\mathbf{x}}[t]$ to denote the remaining symbols that still need detection. This means that the number of unknowns and linear equations that have to be solved is reduced. For proceeding with the decoding of the remaining symbols we need to utilize all the received signals. For proceeding with decoding we also need to express the joint channel matrix that includes the channel gain vectors from all the broadcast and forwarding phases that we defined earlier:

$$\mathbf{H}_d = [\mathbf{h}_{\mathcal{N},d} \quad ... \quad \mathbf{h}_{m,d}]^T \quad (7)$$

Now if $\mathbf{w}_d$ is the vector that contains the noise samples we can write in vector form all the received signals at destination d for a symbol slot t :

$$\tilde{\mathbf{y}}_d[t] = \sqrt{P} \cdot \mathbf{g} \cdot \mathbf{H}_d \cdot \tilde{\mathbf{x}}[t] + \mathbf{w}_d \quad (8)$$

After we use the above description for the signals of interest we can proceed and define the decoding method for the signals that are aligned along the Y dimension.

For a multi-user system the optimal detector is an MMSE-SIC receiver [1]. The Minimum Mean Square Error (MMSE) approach tries to find a coefficient matrix $\mathbf{Q}$ which minimizes the MMSE criterion. We have that $\mathbf{Q}^{\dagger} = (\mathbf{H}^H\mathbf{H}+\sigma^2\mathbf{I})^{-1}\mathbf{H}^H$. So if we would apply directly the MMSE approach it would be:

$$\hat{\mathbf{y}}_d = \mathbf{Q}^{\dagger}_{d,n}\tilde{\mathbf{y}}_d = \mathbf{Q}^{\dagger}_{d,n}\mathbf{H}_{d,n}\mathbf{x} + \mathbf{Q}^{\dagger}_{d,n}\mathbf{w}_d \qquad (9)$$

But in our case we follow an MMSE-OSIC with MRC equalization and detection at the destination node for detection optimality under the proposed protocol. If we denote by $\bar{\mathbf{y}}$ the ordered version from higher to lower power[1] of the received signals contained in $\tilde{\mathbf{y}}$ then we can apply the OSIC approach along the Y dimension. The destination uses MMSE equalization and estimates the higher power $x_n[l]$ symbol (because is the first in array $\bar{\mathbf{y}}$ is denoted as $\bar{\mathbf{y}}_{d,n[1]}$) as

$$\hat{\mathbf{x}}_{d,n[l]} = \mathbf{Q}^{\dagger}_{d,n}\bar{\mathbf{y}}_{d,n[1]} \qquad (10)$$

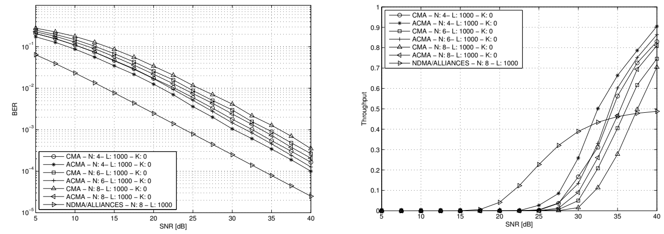
It is important to note that $\hat{\mathbf{x}}_{d,n[l]}$ indicates the estimate of symbols from a source n but at node d.

V. Performance Evaluation

We implemented several systems and we evaluated their performance in terms of BER and throughput under different channel conditions through Monte Carlo simulations. For ACMA we assume that an equal number of broadcast and forwarding phases occurs during a complete transmission round, i.e. N/2 and N/2 respectively. The NDMA/ALLIANCES transmission protocol is also evaluated with the assumption that $L_n = T$ (always perfect synchronization). A protocol named CMA is considered and it uses the forwarding of the sumperimposed signals from $N-1$ relays while the system named RMA uses retransmissions of the same packets from the sources without the involvement of relays. Furthermore, we consider the transmission of packets with different packet sizes which means that the most suitable measure of performance is throughput. For the throughput the correctly decoded packets are accounted for and the results are normalized to 1. We present the averaged results of 1,000 packet transmissions. The channel bandwidth is 20 MHz, the noise over the wireless spectrum is AWGN with the variance of the noise to be 10^{-9} W/Hz at every node/link. We also used a Rayleigh fading wireless channel model. The channel transfer functions between the nodes vary independently but they are characterized by the same average SNR.

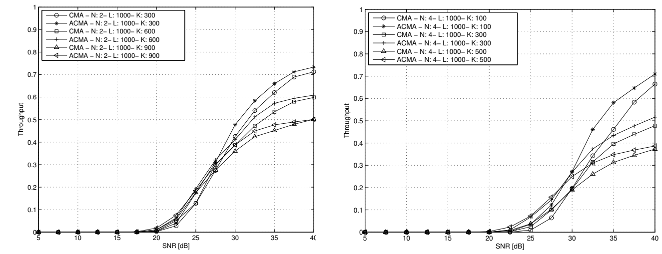
A. Results

Results for multiple sources can be seen in Fig. 3. It is interesting to observe in this figure that as the number of sources is increased, the performance reaches a lower maximum throughput for every scheme. This means that the case of two sources (throughput is not shown for not clogging the presentation) with CMA is the optimal choice under symmetric network conditions and perfect superimposition ($K = 0$). Note the behavior of NDMA/ALLIANCES that is very good in the low SNR regime since a single packet without a collision is transmitted. However, the throughput saturates at a lower value than the remaining schemes since when the channel is improved this class of protocols can use the proposed SIC algorithm and maximize performance. An observation regarding NDMA/ALLIANCES is that it can increase the maximum throughput when the number of tested sources is increased since more collisions occur in this case. Nevertheless, this requires a significant number of additional sources [9]. Another interesting observation has to do with the performance differences between CMA and ACMA. The later always performs better than CMA, while the relative performance gap is increased as the number of nodes is also increased. The reason is that in this case, due to the perfect synchrony assumption, the X-Dimension SIC algorithm is not needed and the under-performance of CMA is because of the noise amplification that occurs at the relays. A higher N means a higher number of used relays and of course higher impact of noise amplification.



(a) BER. (b) Throughput.

Fig. 3. Results for different number of sources N and $K = 0$, $\forall n \in \mathcal{N}$.

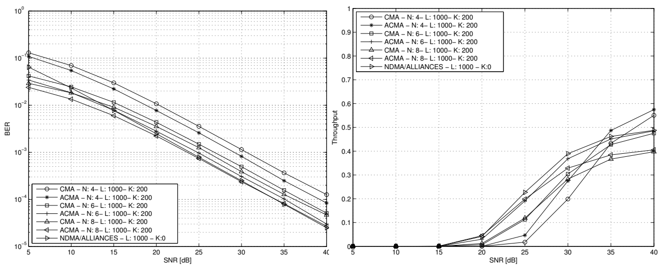


(a) (b)

Fig. 4. Results for two and four sources and different K.

In Fig. 4(a) we see the performance in terms of throughput for different *average* number of free symbol slots K and two sources. As K is increased the average degree of overlap is decreased between different symbols. This has a positive impact on the BER that is reduced since the Y-Dimension SIC and decoding algorithm has to decode fewer superimposed symbols (not shown due to lack of space). However, the BER reduction comes at the cost of lower utilization of the available symbol slots that remain "empty". It appears that this has the highest impact on the final throughput that drops quickly as L_n is increased. Another interesting result in Fig. 4(a) is that as the SNR becomes higher the performance gap between CMA and ACMA is decreased. The reason for this is the

[1] This ordering is easily accomplished through the channel estimation process.



(a) BER. (b) Throughput.

Fig. 5. Results for different number of sources N and $K = 200$ for ACMA and CMA while $K = 0$ for NDMA/ALLIANCES.

reduced impact of noise on the relays at this SNR regime, which makes CMA more efficient. The performance gap is also observed for lower values of K, where significant overlap of symbols occurs. But for the higher values of K and high SNR this gap diminishes. This is because the behavior of the system essentially approaches that of a cooperative diversity protocol with orthogonal transmissions and no symbols are superimposed. In that case, the X-Dimension SIC becomes less relevant and the two systems become practically one and the same. This result also shows that both protocols outperform classical cooperative diversity protocols in the high SNR regime. The results are becoming worse when four nodes broadcast at the same time while the performance drop is faster for an even lower but increasing K as Fig 4(b) indicates. However, the proposed protocols can still provide benefit when $K \leq 300$ and in the high SNR regime when compared to NDMA/ALLIANCES. Also the performance gap between the two protocols in the high SNR regime is higher in this case when compared to the previous experiment of $N = 2$. This occurs precisely because more nodes superimpose their symbols and the number of detected symbols by the X-Dimension SIC is higher. However, again as K is increased the performance of the two protocols tend to converge for the same reasons we explained previously.

We now examine a constant $K = 200$ for all the protocols and different number of sources N in Fig. 5. A first observation we can make from the throughput results in Fig. 5(b) is that there is a minor decrease in the performance difference of ACMA over CMA for lower values of N. The difference becomes less important when N is increased. The reason is again that for higher SNR lower noise amplification occurs. The most interesting results can be observed in the low SNR regime. In this regime the performance differences between the two protocols is attributed to the severe noise amplification that hurts CMA and the error propagation of the proposed SIC algorithm for ACMA. If we look now at the performance regardless of the protocol, both of them underperform when compared (unfairly) to the case of $K = 0$ with NDMA/ALLIANCES. The reason we compare these results is for demonstrating that in a more practical scenario with asynchrony we can still have performance gains. Of course the BER that both protocols can achieve is higher for fewer transmitting nodes contrary to the results for $K = 0$ and NDMA/ALLIANCES. In the low SNR regime the BER is therefore lower for $N = 8$ and so the throughput is higher than the cases of $N = 6$, and $N = 2$. However, the BER performance of both CMA and ACMA protocols converges more in terms of BER for the higher SNR regime leading thus again to the superiority of the system with $N = 2$ that we discussed previously.

VI. Conclusions

In this paper we considered a multiple access scheme that is named ACMA and it exploits over-the-air superimposed packet transmissions through cooperation. The protocol we proposed amplifies and forwards broadcasted packets that are superimposed over-the-air and it also employs additional broadcast/retransmission phases. An essential component of ACMA is a new SIC algorithm that decodes packets of different length that are superimposed asynchronously in the time domain. The goal of ACMA is to increase the system throughput and at the same time remove the requirements for packet scheduling and coordination across the multiple sources. Performance results showed that the proposed protocol performs best when the number of concurrent sources is small but gains are also observed over other protocols when the number of sources is increased while channel quality is improved.

References

[1] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[2] M. Dankberg, M. Miller, and M. Mulligan, "Self-interference cancellation for two-party relayed communication," United States Patent 5596439, January 1997.

[3] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *MobiCom*, 2006.

[4] A. Argyriou and A. Pandharipande, "Cooperative protocol for analog network coding in distributed wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 10, pp. 2014–2023, October 2010.

[5] M. Tsatsanis, R. Zhang, and S. Banerjee, "Network-assisted diversity for random access wireless networks," *Signal Processing, IEEE Transactions on*, vol. 48, no. 3, pp. 702 –711, mar 2000.

[6] R. Lin and A. Petropulu, "A new wireless network medium access protocol based on cooperation," *Signal Processing, IEEE Transactions on*, vol. 53, no. 12, pp. 4675 – 4684, dec. 2005.

[7] J. N. Laneman, D. Tse, and G. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, no. 12, pp. 3062–3080, December 2004.

[8] N. Wu and H. Gharavi, "Linear dispersion codes for asynchronous cooperative mimo systems," in *IEEE ICC*, 2011.

[9] A. Argyriou and A. Pandharipande, "Collision recovery in distributed wireless networks with opportunistic cooperation," *IEEE Communications Letters*, vol. 14, no. 4, pp. 300–302, April 2010.