A Threshold Based MAC Protocol for Cooperative MIMO Transmissions

Haiming Yang, Hsin-Yi Shen, Biplab Sikdar and Shivkumar Kalyanaraman
Department of ECSE, Rensselaer Polytechnic Institute,
Troy, NY 12180 USA

***Abstract*— This paper develops a distributed, threshold based MAC protocol for cooperative Multi Input Multi Output (MIMO) transmissions in distributed wireless systems. The protocol uses a thresholding scheme that is updated dynamically based on the queue length at the sending node to achieve low power transmissions while ensuring stability of the transmission queues at the nodes. Simulation results are provided to evaluate the performance of the proposed protocol and compare it against regular point to point as well as fixed group size cooperative MIMO MAC protocols.**

I. Introduction

MIMO techniques achieve the same bit error rate (BER) as point to point communications by allowing nodes to transmit and receive information jointly with lower power [1], thereby achieving lower overall energy consumption and higher channel capacity in wireless fading channels. Since wireless sensor networks (WSNs) typically consist of a large number of energy constrained sensor nodes with limited on board battery resources which are difficult to recharge or replace, MIMO techniques can be used to save power. However, using multi-antenna techniques directly in WSNs is impractical since the limited size of a sensor node usually supports a single antenna. Cooperative MIMO schemes have thus been proposed for WSNs to improve communication performance [2]. Cooperative transmissions and receptions from antennas in a group of sensor nodes can be used to construct a system fundamentally equivalent to a MIMO system. The cooperative MIMO system spends more energy on exchanging cooperation control messages but higher energy saving is achieved during the long-haul data transmissions.

The complexity of coordinating the actions of distributed nodes limits the practical use of cooperative MIMO in WSNs. Also, an inefficiently designed MAC protocol will increase the energy spent in exchanging control messages, and diminish the performance gains of MIMO operation. Another important factor that determines the effectiveness of the cooperative transmission strategy is the tradeoff between energy savings and stability. While scheduling transmissions only when a large number of cooperating nodes are available improves the energy savings, it also increases the likelihood that the queue at the sender becomes unstable. Thus a MAC protocol must dynamically select the cooperating group size based on the network conditions.

To address these issues and facilitate cooperative MIMO transmissions with a high degree of performance improvement, this paper proposes a new MAC protocol. In the proposed protocol, packets are cooperatively transmitted by a node only when the transmission BER is expected to achieve a dynamically set threshold. This cooperative threshold is set to achieve the maximum throughput while maintaining the stability of the queue at the sender and is dynamically changed according to the existing network conditions. Furthermore, the numbers of nodes in the sending and receiving groups are set to achieve the minimum energy consumption subject to satisfying the cooperative threshold. The proposed protocol outperforms point to point communications as well as cooperative MIMO MAC protocols that use fixed groups sizes [6].

The rest of the paper is organized as follows. Section II describes the related work and section III presents the proposed MAC protocol. Simulation results to evaluate and compare the performance of the proposed protocol are presented in Section IV. Section V concludes the paper.

II. Related Work

Most of the existing MAC protocols for MIMO systems focus on MIMO transmissions from a single node and not collaborative transmissions [7], [8]. Existing MAC protocols for collaborative MIMO transmissions are based on centralized, cluster based architectures [3]. Centralized architectures lead to energy wastage on cluster maintenance, introduce additional coordination delays and present a central point of failure.

A cooperative MIMO MAC protocol has been proposed in [6] that forms sending and receiving groups at the time of each packet transmission. The transmission system achieves lower overall energy consumption than point to point communications. However, the numbers of nodes in the sending and receiving groups are fixed and it is difficult to set the right numbers for the groups to achieve the minimum energy consumption and may cause the system to become unstable.

In contrast to existing work, we propose a throughput optimal, distributed MAC protocol with thresholding to control the decision to proceed with a transmission. The protocol is easy to deploy and is shown to perform better than point to point and fixed groups size cooperative MIMO MAC protocols.

III. Protocol Description

The proposed cooperative MIMO communication strategy consists of three steps, as shown in Figure 1. In the first step, the source broadcasts its data using low transmission power to the source cluster members (Figure 1(a)). In the

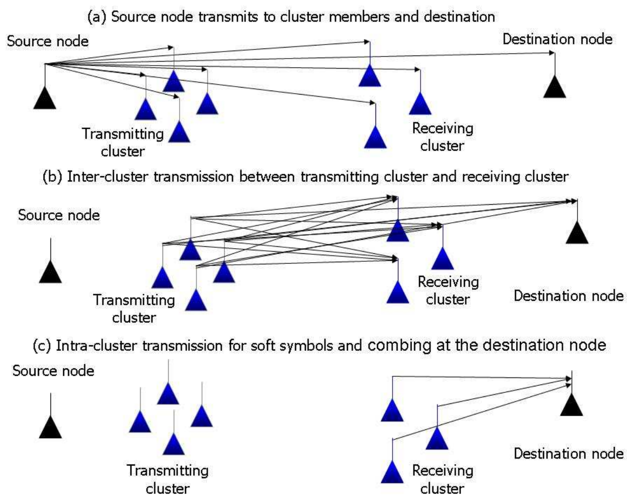

Fig. 1. Proposed cooperative scheme: (a) Source node sends information to all cluster members and destination, (b) Inter-cluster transmission, (c) Receiving nodes sequentially relay signal copies to destination node and soft symbol combining at destination.

second step (Figure 1(b)), nodes in the source-cluster relay the signals received in the first step to the destination-cluster. In the third step (Figure 1(c)) each receiver sequentially transmits the received data to the destination node after passing it through a decision feedback equalizer (DFE) and soft symbol decoding. The data transmission in this cooperative MIMO structure with $M \times N$ nodes in the source and destination clusters can be treated as several MISO systems and error combining of N packet copies [5]. The data transmission error can be dramatically reduced by using multiple senders in the sending side and error combining in the receiving group. We next describe the proposed MAC protocol for coordinating the multi-step transmissions from multiple nodes, as required by the cooperative MIMO communication system.

A. MAC Protocol

We first consider the operation of a node that generates or receives a packet to be forwarded. Before beginning the transmission process, the node first senses the channel to ensure that it is idle, as in any CSMA mechanism. If the channel is sensed to be busy, the node initializes a backoff timer (as in IEEE 802.11) in the range $[1, CW_{min}]$ and waits for the channel to be idle. The timer is decremented once the channel is sensed idle and interrupted if the channel becomes busy again. If the channel is idle and the backoff counter has decremented to zero, the source node first broadcasts a message with low transmission power (at most half the normal transmission power) to its local neighbors in an attempt to recruit them for transmitting the packet. Once replies have been received from the neighbors, the source node sends out a RTS message to the destination at normal power to reserve the channel and waits for the CTS reply. The RTS message contains information on the current queue length at the sender and the number of neighbors it has recruited. This information is used by the receiver to update the cooperative threshold.

If the source does not receive a CTS packet within an interval T_{int}, a retransmission process begins. The retransmission process is based on a binary exponential backoff mechanism as in IEEE 802.11 and upper bounds of $RxLimit$ and CW_{max} are placed on the number of retransmission attempts and the maximum contention window size. Also, if the source node receives a negative CTS (NCTS) packet from the destination node (implying that the cooperative threshold condition is not met) the source will backoff and attempt a retransmission, starting with the recruitment of cooperative nodes.

Once a CTS packet is received in response to its RTS packet, the source node proceeds with the data transmission. Each CTS packet contains the optimal size of the cooperating group at the sending side (the process is described in Section III-C). T_{int} seconds after receiving a CTS packet, the source node broadcasts the data packet at low power to the nodes in its group and synchronizes them. The protocol specifies that each node in the source-cluster transmit its data exactly T_{syn} seconds after the broadcast data packet is received from the source node. Given that the distances between the source and each of its cluster members is expected to be less than 100 meters, the differences in their propagation delays is quite small. Our receiver design then uses a DFE and soft-symbol combining to correct for the asynchrony [9]. Once the packet has been cooperatively transmitted, the source waits for an ACK from the destination. The amount of time it waits for the ACK depends on the product of the number of nodes N in the destination cluster and the channel data rate. If no ACK is received the retransmission process begins after a binary exponential backoff starting from neighbor recruitment.

We next consider the operation of the destination node. On receiving the RTS packet, if the channel around it is idle, after an interval of T_{int} seconds the destination sends out its own recruiting packet, at a power level at most half that of the normal value. On receiving the replies from nodes willing to cooperate, the destination node uses this information along with the number of available cooperating nodes and the queue length at the sender (this information is passed through the RTS packet) to calculate the threshold. The methodology to calculate the threshold is described in Section III-B. This threshold BER value is then compared with the estimated BER value of the channel between the source and destination node (using the recruited cluster sizes and the interference plus noise value of the received RTS signal).

If the channel's estimated BER is higher than the cooperative threshold, a negative CTS (NCTS) packet is sent to the sender to cancel the transmission. On the other hand, if the threshold is met, the destination node first calculates the size of the sending and receiving groups (subject to the threshold BER and available cooperating nodes) that achieves the minimum overall energy consumption (the process is described in Section III-C), and broadcasts a low power message to notify the nodes who have been selected to help in the reception. This message also includes the order in which the cooperating nodes relay the data packet after the transmission from the source cluster. It then sends a CTS packet with the required cooperating group size to the source node and waits for the data packet. The destination node then waits for the data transmission from the source cluster. Next, it waits for each node in the destination cluster to sequentially forward its copy of the received data packet. Finally, it decodes the packet by combining all copies of the received packet and replies with an ACK packet to the source if the packet is decoded correctly. Otherwise, the destination node does nothing and the source node will eventually timeout.

We next consider the operation of cooperating nodes that form the source and destination clusters. Each node that receives the recruiting message from the source node and is not constrained by any other transmissions in its vicinity, may opt to help with forthcoming transmission. Each node that decides to cooperate with the transmission replies to the source node with an ACK packet. To avoid collisions, each node generates a random backoff value in the range $[0, CW_{min}]$ and follows the usual backoff mechanism before transmitting the ACK. Collisions may still occur between the replying nodes, but in the interest of reducing the overall transmission time, no retransmissions are attempted.

Each node that receives the recruiting message from a destination node and is available for helping with the forthcoming transmission, replies to the destination node with an ACK. Again, this ACK is sent after a random backoff in order to avoid collisions. After transmitting the ACK, each node waits for the message from the destination node to confirm its selection in the destination cluster. If a node is not included in the cluster, it goes back to its normal operation and takes no further part in the forthcoming transmission. On the other hand, if it is included in the destination cluster, each node waits for the data transmission by the source cluster. On receiving the data, each node uses a DFE to correct the asynchrony (as described in [9]) and then forwards the data to the destination in the order specified.

B. Thresholding Policy

In this section we propose a thresholding policy Π_O for the proposed MAC protocol. The decision to transmit or not, along with the choice of the cluster size if a transmission is attempted, is based on the queue length at the source node as well as the availability of cooperating nodes.

Let the maximum number of nodes available for cooperation with the source and destination nodes be M_{max} and N_{max} respectively. Including the source and destination nodes, there are thus $M_{max} + 1 \times N_{max} + 1$ possible choices for the tuple M, N denoting the source and destination cluster sizes. For each possible choice of M, N, the expected packet error rate (PER) for the given channel noise conditions, $P_e[M, N]$, is first evaluated. Let the number of unique PERs be K. The successful packet transmission probability for each case is obtained by subtracting the PER from one and these K probabilities of successful transmission are then listed in ascending order $\varphi(1), \varphi(2), \cdots, \varphi(K)$. Different cluster size combinations that lead to the same PER are thus considered a single entry in the list. Also, let $\phi(i)$ be the mapping $i \to M, N$ from the successful transmission rates to the cluster sizes:

$$\phi(i) = \{(M, N) \mid 1 - P_e(M, N) = \varphi(i)\} \tag{1}$$

The policy Π_O chooses the threshold in the following way. If the current queue length at the sender is Q, threshold i (i.e. $\varphi(i)$ in terms of the desired successful packet transmission probability) is chosen if $(K - i)\xi < Q \leq (K - i + 1)\xi$ where ξ is a fixed positive integer. The threshold is set at 1 for $Q > K\xi$. The threshold i chosen by the policy is mapped using Eqn. (1) to obtain the set S of cluster sizes for which the packet delivery rate is greater than $\varphi(i)$: $S = \{(M, N) \mid 1 - P_e(M, N) \geq \varphi(i)\}$. Let the number of nodes, including the sender and receiver, that have been recruited to help with a given transmission be M_a and N_a respectively. The sender proceeds with the transmission only if $(M_a, N_a) \in S$.

C. Node Selection in Sending and Receiving Groups

Consider a cooperative MIMO transmission with M source and N receiving nodes. In our protocol, all cooperative sending nodes transmit at the same power and synchronously send the same data. A combination of these transmissions is detected at each receiving node, equivalent to a MISO scenario. The error rate at a receiving node $p_{e_{M\_1}}$ is thus related to the power summation from multiple signal transmission paths. In addition to the $p_{e_{M\_1}}$ in each route, the error from a receiving node to the destination $p_{e_{pp}(dst)}$ and the error from the source to any node in the sending group will also contribute to the overall route error. The latter two errors are quite low and are neglected in our analysis and we assume $p_e = p_{e_{M\_1}}$.

The destination node uses a simple majority decision rule to decode the packets received from the cooperating receivers. If each node in the receiving group has the same BER, the BER in the destination node after the reception from the N nodes forming the reception group is:

$$p_b = p_{e_{M\_N}} = \sum_{i=N/2}^{N} \binom{N}{i} p_e^i (1 - p_e)^{N-i} \tag{2}$$

For cooperative MIMO, the error rate in a Rayleigh Fading channel using BPSK modulation without channel coding is given by [6]

$$p_e \quad = \quad \prod_{i=1}^{M} \frac{1}{1 + \frac{P_t(mimo)}{N_0 d_{ij}^{\alpha}}} \tag{3}$$

where $P_t(mimo)$ is the transmission power at each node in the sending group, N_0 is the ambient noise and d and λ are the distance and fading gain from the sending node to the receiving node. The path loss constant α is between 2 and 4. Using the value of p_e from Eqn. (3) in Eqn. (2), the overall BER at the destination node is given by

$$p_b = \sum_{i=N/2}^{N} \binom{N}{i} \left(\prod_{i=1}^{M} \frac{1}{1+ \frac{P_t(mimo)}{N_0 d_{ij}^{\alpha}}} \right)^i \left(1 - \prod_{i=1}^{M} \frac{1}{1+ \frac{P_t(mimo)}{N_0 d_{ij}^{\alpha}}} \right)^{N-i} \tag{4}$$

If no FEC codes are used, the packet error rate p_p is:

$$p_p = 1 - (1 - p_b)^L \tag{5}$$

where L is the frame length in bits.

The energy consumed by a sensor node consists of two parts: energy spent on running the circuits P_c and the transmission energy P_t spent on communications. The same energy is spent on running the circuits irrespective of whether the node is transmitting, receiving or idle listening [4]. Since the circuit power consumption is independent of whether a node is transmitting or not, we only consider the transmission energy for evaluating the energy cost of a transmission.

The energy consumption for each transmission can be divided into two parts: the energy spent on channel reservation and recruiting (E_{wait}) and the energy spent on data transmission ($E_{trans\_M}(i)$), which also depends on the chosen threshold i (or $\varphi(i)$). Now, E_{wait} includes the RTS/CTS exchange as well as the recruitment process and is given by

$$E_{wait} = E_{rts} + E_{cts} + E_{recruit} \tag{6}$$

where E_{rts} and E_{cts} are the energy spent on sending RTS and CTS packets and $E_{recruit}$ is the energy spent on recruiting neighboring nodes. Assuming all the neighboring nodes reply to the recruiting messages and that recruiting messages and their replies require energies of E_{rec_s} and E_{rec_d}, we have

$$E_{recruit} = 2E_{rec\_s} + (M_{max} + N_{max})E_{rec\_d} \tag{7}$$

With M nodes in the sending and N nodes in the receiving group, the energy consumption in a successful transmission is

$$E_{s\_M} = E_{wait} + E_{B_S} + E_{B_R} + M E_{data\_M} + (N-1)E_{col} + E_{ack} \tag{8}$$

where E_{B_S} is the energy spent by the source to send the data to its cooperative neighbors, E_{B_R} and E_{ack} are the energy spent by the destination to send notification messages to its recruited neighbors and the ACK to the source, E_{data_M} is the energy spent by a source-cluster node for transmitting data and E_{col} is the energy spent while the destination collects the message from cooperating receivers. In case the transmission is unsuccessful, the energy consumed is

$$E_{u\_M} = E_{wait} + E_{B_R} + E_{B_S} + M E_{data\_M} + (N-1)E_{col} \tag{9}$$

With the PER, p_p, given in Eqn. (5), the number of attempts before a successful transmission follows a geometric distribution. Combining the successful and unsuccessful cases gives:

$$E_{trans\_M}(i) = \frac{p_p}{1 - p_p} E_{u\_M} + E_{s\_M} \tag{10}$$

Group Size Selection: The destination node in the proposed protocol determines the sending and receiving group sizes based on the framework above. Given the number of nodes available at the sending and receiving groups, the destination node does an exhaustive search of the possible sizes and selects the combination that has the lowest energy consumption, subject to the threshold. The exhaustive search is necessary since there is no closed form solution of Eqn. (10) that achieves the minimum value. Since the number of available nodes is typically small, the computational complexity is acceptable.

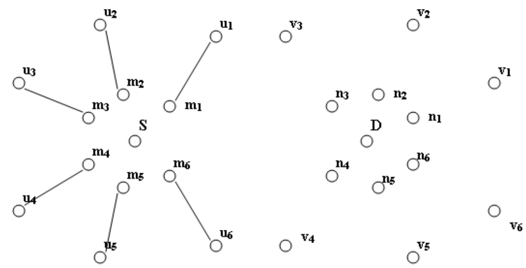

Fig. 2. Topology

IV. Simulation Results

In this section, we evaluate the performance of the proposed MAC protocol using a custom built MATLAB simulator. In order to compare the performance fairly, we set the overall transmission powers of all protocols to be the same. So if the transmission power of each sending node in a 1×1 network is P_t, the transmission power of each sending node in 2×2 and 4×4 networks will be $P_t/2$ and $P_t/4$ respectively.

The topology used for the simulations is shown in Fig. 2. We are interested in the transmissions between the source and destination nodes marked S and D. Around both S and D, there are six nodes that may serve as cooperative nodes and these are marked $m_1, \cdots, m_6$ and $n_1, \cdots, n_6$. These twelve nodes also exchange data with their neighboring nodes using unicast flows and thus may not always be available for cooperative transmissions. Each unicast sender U_i generates packets at rate λ_U according to a Poisson process. Also, S generates packets at rate λ_S for destination D. The cooperative threshold is initially set at 0.1 and is then dynamically updated.

We use UDP as the transport layer. To compare the performance of the protocols, we measure the overall energy consumption given by the sum of the transmitting power used by all the nodes participating in the transmission. Each simulation is run for 6000 seconds. The channel has capacity 1 Mbps. The size of RTS, ACK and recruiting messages at source and destination is 44bytes. The size of CTS, notification message from the destination node and reply messages from helping nodes is 38 bytes. The size of data messages is 256 bytes. The recruitment messages use one-fourth the power used for data transmissions.

We compare the performance of the proposed protocol with three others: traditional point to point, 2×2 (fixed threshold) cooperative MIMO and 4×4 (fixed threshold) cooperative MIMO transmissions. For the entire range of transmission powers used in our results, the packet error rate with point to point transmissions was greater than 0.7, leading to excessive retransmissions and thereby higher delays and lower throughputs. Thus the results for point to point transmissions are not shown in any of the figures.

A. Simulations Without Neighboring Traffic

When none of the neighboring nodes have any traffic of their own, the source and destination nodes can always recruit

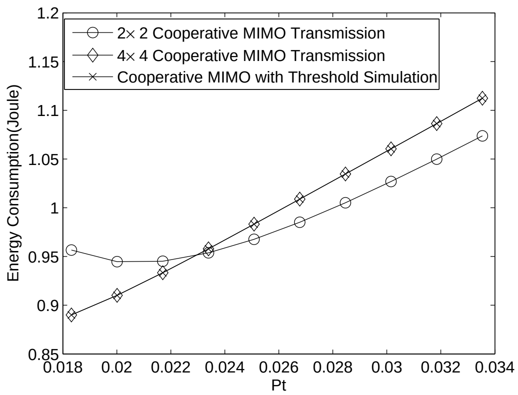

Fig. 3. Energy consumption under different transmission powers.

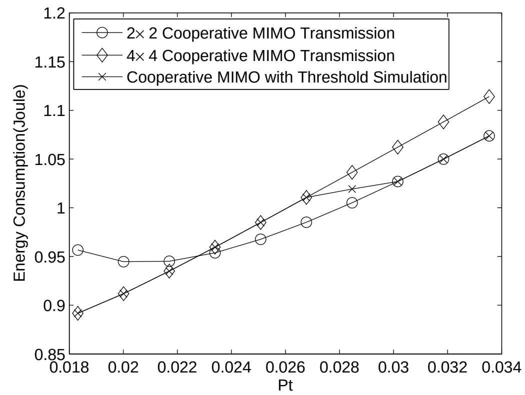

Fig. 5. Energy consumption in the presence of neighbor traffic.

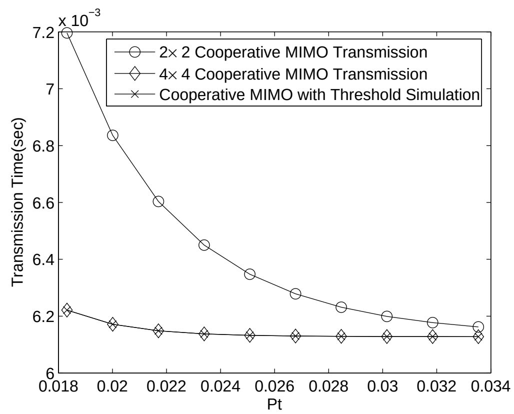

Fig. 4. Packet delay under different transmission powers.

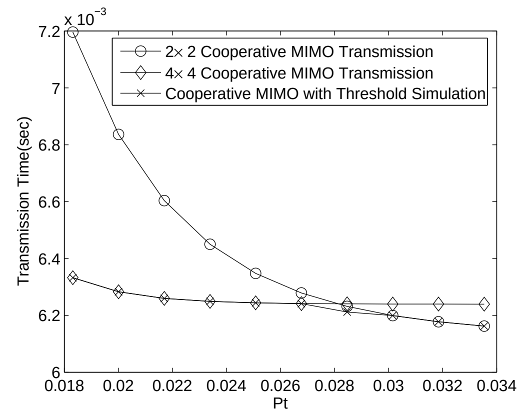

Fig. 6. Packet delay in the presence of neighbor traffic.

all the neighboring nodes and there is no contention for the channel. The proposed MAC protocol thus chooses the highest level cooperative threshold with probability 1. The energy consumption and packet delays for the various protocols are shown in Figures 3 and 4 respectively, for $\lambda_S = 0.32$.

Figure 3 shows that the proposed threshold based MAC protocol leads to significant energy saving at low transmission powers. Figure 4 compares the delays associated with the protocols. The proposed protocol has relatively much lower delays at lower transmission powers.

B. Simulations with Background Traffic

In this set of simulations, we introduce traffic at rate $\lambda_U = 0.7$ at each partner of the cooperating neighbor nodes while the source node generates traffic at rate $\lambda_S = 0.65$. Thus the source and destination nodes may not always be able to proceed with their transmission or recruit all nodes. The energy consumption and packet delays for this scenario are shown in Figures 5 and 6. We observe that the proposed MAC protocol outperforms fixed threshold MAC protocols by changing the cooperative threshold according to the queue length, thereby reducing the energy spent on recruiting and the time spent on waiting for the required number of nodes in the transmission.

V. Conclusion

This paper presents a new cooperative MIMO MAC protocol with dynamic thresholding for WSNs. Transmissions in the proposed protocol proceed only when the expected transmission BER is lower than the cooperative threshold and sending and receiving group sizes are selected to achieve the minimum energy consumption.

References

[1] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, ”Space-time block codes from orthogonal designs,” *IEEE Transactions on Information Theory,* vol. 45, no. 5, pp. 1456-1467, 1999.

[2] S. Cui, A. J. Goldsmith, and A. Bahai, ”Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks,” *IEEE Journal on Sel. Areas in Commun.,* vol. 22,no. 6, pp. 1089-1098, 2004.

[3] Y. Yuan, M. Chen, and T. Kwon, ”A Novel Cluster-Based Cooperative MIMO Scheme for Multi-Hop Wireless Sensor Networks,” *EURASIP Journal on Wireless Communications and Networking,* 2006.

[4] M. Stemm and R. Katz, ”Measuring and Reducing Energy Consumption of Network Interfaces in Hand-held Devices,” *IEICE Transactions on Communications,* Aug. 1997.

[5] S. Yi, Y. Shan, S. Kalyanaraman and B. Azimi-Sadjadi, ”Header Error Protection for Multimedia Data Transmission in Wireless Ad Hoc Networks,” *Proceedings of IEEE ICIP,* October 2006.

[6] H. Yang, H. Shen, and B. Sikdar, ”A MAC Protocol for Cooperative MIMO Transmissions in Sensor Networks,” *Proc. IEEE GLOBECOM,* 2007.

[7] D. Hoang and R. Iltis, ”An Efficient MAC Protocol for MIMO-OFDM Ad hoc Networks,” *Proc. of IEEE Asilomar Conference,* October 2006.

[8] K. Sundaresan, R. Sivakumar, M. Ingram and T. Chang, ”Fair Medium Access Control Protocol for Ad-hoc Networks with MIMO Links,” *Proceedings of IEEE Infocom,* March 2004.

[9] H.-Y. Shen, H. Yang, B. Sikdar and S. Kalyanaraman, “A Distributed System for Cooperative MIMO Transmissions,” *Proc. IEEE GLOBECOM,* 2008.