Clustering and Power Management for Virtual MIMO Communications in Wireless Sensor Networks

Marwan Krunz[1] and Mohammad Z. Siam[2] and Diep N. Nguyen[1]
[1]Department of Electrical and Computer Engineering, University of Arizona, USA
E-mail:{krunz, dnnguyen}@email.arizona.edu
[2]Al-Isra University, Amman, Jordan
E-mail:{siam}@email.arizona.edu

Abstract

Multi-input multi-output (MIMO) is a well-established technique for increasing the link throughput, extending the transmission range, and/or reducing energy consumption. In the context of wireless sensor networks (WSNs), even if each node is equipped with a single antenna, it is possible to group several nodes to form a *virtual antenna array*, which can act as the transmitting or receiving end of a virtual MIMO (VMIMO) link. In this paper, we propose energy-efficient clustering and power management schemes for virtual MIMO operation in a multi-hop WSN. Our schemes are integrated into a comprehensive protocol, called cooperative MIMO (CMIMO), which involves clustering the WSN into several clusters, each managed by up to two cluster heads (CHs); a *master CH* (MCH) and a *slave CH* (SCH). The MCH and SCH collect data from their cluster members during the intra-cluster communications phase and communicate these data to neighboring MCHs/SCHs via an inter-cluster VMIMO link. CMIMO achieves energy efficiency by proper selection of the MCHs and SCHs, adaptation of the antenna elements and powers in the inter-cluster communications phase, and using a cross-layer MIMO-aware route selection algorithm for multi-hop operation. We formally establish the conditions on the transmission powers of CHs and non-CHs that ensure the connectivity of the inter-cluster topology. Simulations are used to study the performance of CMIMO. The simulation results indicate that our proposed protocol achieves significant reduction in energy consumption and longer network life time, compared with non-adaptive clustered WSNs.

Index Terms

Virtual MIMO, wireless sensor networks, clustering, routing, MAC protocol, power control, cooperative communications.

I. INTRODUCTION

Nodes in wireless sensor networks (WSNs) are typically powered by small batteries. Replacement or recharging of these sensors is often difficult due to two reasons: (1) Sensors are deployed in large numbers, making the process of manually recharging them expensive and time consuming, and (2) in some applications, such as relief-and-rescue and battlefields, it may be infeasible to reach the sensors once they have been dispatched. Consequently, improving the energy efficiency of a WSN has always been a primary design objective.

A preliminary version of this paper was presented at the *IEEE INFOCOM 2009 Conference*, Rio de Janeiro, Brazil, April 2009 [1]. This research was supported in part by NSF (under grants CNS-1016943, CNS-0904681, IIP-0832238), Raytheon, and the Connection One center (an I/UCRC NSF/industry/university consortium). Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Multi-input multi-output (MIMO) technology has the potential to increase the throughput and/or reduce the transmission energy consumption. This is done by exploiting three types of gains: Array, multiplexing, and diversity [2]. Array (or power) gain is achieved at the receiver through coherent combining of multiple copies of the signal. Multiplexing gain is obtained when different signals are transmitted simultaneously over several antennas to increase the total transmission capacity of a link. Diversity gain, achieved by sending/receiving several highly correlated versions of a signal over independent fading paths (different antennas), is the slope of the average bit error rate (BER) curve versus signal-to-noise ratio (SNR). In this work, we focus on diversity gain, leaving the exploitation of other types of gain for future research.

A typical MIMO system requires multiple antennas at the transmit and/or receive end of a link. However, in dense topologies such as WSNs, it is also possible to group two or more single-antenna nodes to form a cooperative (virtual) multi-antenna node. Forming such a virtual node requires sensor nodes to exchange information and decide on the data to be transmitted cooperatively. To ensure that the energy overhead of information exchange is manageable, only those nodes that are geographically close to each other should be part of the virtual node.

In general, data obtained by sensor nodes in dense WSNs exhibit a high degree of redundancy, which can be significantly reduced by means of aggregation/fusion [3]. Data aggregation is facilitated by node clustering, which organizes the network into a connected hierarchy [4]. In the context of WSNs, clustering involves grouping nodes and electing a cluster head (CH) such that the non-CH nodes of a cluster can directly communicate with their CH. CHs forward aggregated data to the sink directly or via other CHs. Topologically, the collection of CHs in the network forms a connected dominating set.

In this paper, we propose a distributed MIMO-adaptive energy-efficient clustering/routing protocol, coined cooperative MIMO (CMIMO), for multi-hop WSNs. According to this protocol, each cluster has up to two CHs, which are responsible for inter-cluster communications. Clustering is done based on the remaining battery lifetime (RBL), neighbor proximity, and network density. The rationale for adopting these criteria is to construct cooperative MIMO links whose effect is as close as possible to actual MIMO systems (with two antennas per node) and that have manageable overhead. The diversity gain of such a cooperative MIMO system is maximized by adapting the "transmission mode" and the transmission power of the inter-cluster virtual MIMO link on a per-packet basis. By "transmission mode" we mean one of four possible transmit/receive configurations: 1×1 (single-input single-output

/ SISO), 2×1 (multi-input single-output / MISO), 1×2 (single-input multi-output / SIMO), and 2×2 (multi-input multi-output / MIMO)[1]. To reduce energy consumption, CMIMO considers both transmission and circuit energies. For a given target BER, a multi-antenna transmission requires less transmission power than a SISO system. However, it also requires more circuit power at both ends of the link. As a result, a distance-dependent tradeoff emerges between transmission and circuit powers [6]: For relatively small distances, circuit power is dominant, and hence a SISO mode is more energy-efficient than a multi-antenna mode. As the transmitter-receiver distance increases, the tradeoff shifts in favor of multi-antenna modes (SIMO, MISO, MIMO). Once the optimal configuration is determined for each inter-cluster link, CMIMO executes an energy-efficient routing (EER) algorithm on the virtual topology to determine the "optimal" end-to-end path that minimizes the total energy consumption.

It should be noted that although this work focuses on clusters with at most two CHs per cluster, the proposed methodology is actually applicable to any number of CHs. Specifically, the proposed procedure and criteria for network clustering and coordination of the cooperation process do not depend on the number of cooperating nodes. However, optimizing this number (and selection) leads to a combinatorial problem of high computational complexity. Thus, to maintain a reasonable computational overhead, we limit our treatment to two CHs per cluster.

The rest of the paper is organized as follows. Section II provides related work. We describe the CMIMO protocol in Section III. The system model and the energy consumption analysis are provided in Section IV. In Section V we discuss some issues related to the design of CMIMO, including connectivity, synchronization, reclustering, and medium access control. The performance of the proposed protocol is evaluated via simulations in Section VI. Section VII discusses the main conclusions of this paper as well as some generalizations and extensions.

II. RELATED WORK

In this section, we describe recent works on VMIMO systems and node clustering in WSNs. VMIMO was first proposed by M. Dohler in [7] [8] in the form of virtual antenna arrays. Then, several VMIMO systems for WSNs have been proposed in the literature (e.g., [6], [9], [10]). In these systems, several single-antenna nodes cooperate on information transmission/reception to achieve energy-efficient communications. The authors in [6] studied a cooperative MIMO scheme with Alamouti code for *single-hop* transmissions in WSNs. They analyzed the best

[1]The feasibility of adapting the transmission mode on a per-packet basis was demonstrated in several experimental MIMO platforms (e.g., [5]).

modulation and transmission strategy to minimize the total energy consumption required to send a given number of bits. The results showed that over certain distances, both the total energy consumption and the total delay can be reduced, even when the energy and delay costs associated with the local information exchange are taken into account.

A cooperative MIMO scheme for delay and channel estimation was proposed in [9]. This scheme uses two transmitting sensors and space-time block codes to provide transmission diversity in distributed WSNs. Full diversity and full rate were achieved, which enhance power/bandwidth efficiency and reliability. It should be noted that neither antenna arrays nor transmission synchronization were used. In [10] energy efficiency and training overhead of cooperative MIMO WSNs were analyzed. The author compared the performance of such systems with that of SISO-based WSNs. The dependence of energy efficiency on the coherence time of the fading process and on the communications distance was considered. The incorporation of data aggregation into cooperative MIMO was recently considered [11]. Multi-hop VMIMO communications with distributed space-time coding were investigated in [12].

All the above VMIMO schemes exploit the diversity gain using distributed space-time codes. In [13] the author exploited the multiplexing gain of VMIMO to reduce the cooperation overhead and the circuit energy consumption. VMIMO operation was realized by using the Vertical-Bell Laboratories Layered Space-Time (VBLAST) technique. The scheme in [13] also conserves significant energy, compared with SISO-based schemes. Using VBLAST for spatial multiplexing, the authors in [14] optimized MIMO's operation under power and delay constraints.

In all the above-mentioned contributions, clustering and multi-hop routing were not taken into consideration, which limits the scalability of these schemes in large WSNs. The authors in [15] argued that by jointly considering the clustering and routing problems, one can reduce the signaling overhead of the routing task, hence conserving significant energy.

Many clustering schemes were proposed for WSNs, which can be classified based on two criteria [16]: (1) The parameters used for electing CHs, and (2) the execution nature of the clustering algorithm (probabilistic or iterative). Some clustering schemes under the first category use the node ID to elect CHs. Others favor nodes with larger degrees. Some other schemes were proposed for controlling the network topology by exploiting node redundancy. Regarding the second category, the execution of a clustering scheme can be carried out at a centralized authority (e.g., a base station) or in a distributed way at local nodes. In iterative clustering schemes, a node waits for a specific event to occur or certain nodes to decide their role (e.g., become CHs) before making a decision. This results in some delay in the convergence time. On the other hand, probabilistic (or randomized) clustering schemes ensure rapid convergence while achieving some favorable properties, such as balanced cluster sizes.

DCA [17] is one of the popular clustering schemes, which clusters nodes in an iterative way. In DCA, nodes divide themselves into groups according to a weight-based criterion. The main assumptions behind DCA are: (1) The network topology is static, and (2) each transmitted message is correctly received by all neighbors within a specific duration of time. As the first assumption is reasonable for WSNs, the second one opens several issues with respect to reliability and collisions.

HEED [18] is a clustering scheme that does not make any assumptions about the presence of infrastructure or about node capabilities, other than the availability of multiple power levels in sensor nodes. The key idea behind this scheme is to periodically select CHs according to a hybrid metric that combines the node's RBL and a secondary parameter, such as node degree. The authors showed that with appropriate bounds on node density and intra/inter-cluster transmission ranges, HEED can asymptotically almost surely guarantee connectivity of clustered networks.

The authors in [19] developed a low-energy adaptive clustering hierarchy (LEACH), which combines the ideas of energy-efficient cluster-based routing and medium access together with application-specific data aggregation, so as to improve the performance in terms of network lifetime, latency, and application-perceived quality. LEACH enables self-organization of large numbers of nodes, adapts clusters, and rotates CH role to evenly distribute the energy load among all the nodes.

Clusters formed under protocols like LEACH, HEED, and DCA have identical sizes, i.e., the number of nodes or the CH ranges are roughly the same for all clusters. The authors in [20] and [21] proposed clustering protocols that result in different cluster sizes. The optimal cluster size problem was formulated in [20] while taking into account the spatially correlation in data. To achieve higher energy efficiency, the optimal cluster size in [20] increases with the relative distance from the cluster to the sink. The primary goal of [21] is not to improve the energy efficiency but to better balance energy consumption between CHs to indirectly improve the network lifetime. The rationale behind [21] is that CHs around the sink have to relay more traffic than outsider CHs, hence faster depleting their batteries. When nearby-the-sink CHs stop functioning, the network may get disconnected. In addition to the inter- and intra-cluster ranges, authors of [21] allow nodes around the sink to use a shorter range to cluster. This results in smaller cluster size for cluster surrounding the sink.

In contrast to all earlier mentioned protocols whose clusters are disjoint (i.e., a sensor cannot belong to two neighboring clusters), [22] clusters the network so that neighboring clusters overlap at different level. This is to facilitate inter-cluster routing, node localization, and time synchronization protocols. The problem of joint coverage, routing, and clustering optimization was visited in [23]. CHs are selected so as to maximize the network lifetime while guaranteing full coverage and network connectivity.

It should be noted that all of these clustering schemes do not exploit cooperative MIMO with clustering. Moreover, some of these schemes do not consider multi-hop communications and early data aggregation to save energy. On the other hand, CMIMO is a cooperative-MIMO based scheme, where the decisions of the nodes that follow CMIMO are based on MIMO-related parameters with multi-hop operation and early data aggregation.

The authors in [24] extended the LEACH scheme to build a cluster-based cooperative MIMO scheme for WSNs, namely MIMO-LEACH. The main differences between CMIMO and MIMO-LEACH can be summarized as follows. First, MIMO-LEACH is based on an existing clustering protocol (LEACH) that does not take into account MIMO operations in the clustering process, whereas CMIMO has its own clustering protocol that exploits MIMO operations in its selecting criteria. Second, MIMO-LEACH contains one CH per cluster, which is responsible for aggregating data and broadcasting it to two other "cooperative" nodes. These nodes are responsible for forwarding the data to the CH of a receiving cluster. It is clear that in [24], three nodes (one CH and two cooperative nodes) are needed to aggregate and forward the data, whereas only the two CHs (MCH and SCH) are the ones that perform these functions in CMIMO. As a result, MIMO-LEACH involves more overhead than CMIMO. Gong et al. extend the work in [1] under the multi-channel scenario in [25]. Finally, four transmission modes are available in CMIMO for each link between any two clusters. However, in MIMO-LEACH no complete MIMO communications (2×2) exist among the network, as the best transmission mode that can be used is SIMO/MISO.

III. THE CMIMO PROTOCOL

A. Overview

CMIMO is a distributed protocol for clustering and virtual MIMO communications that aims at minimizing the total energy consumption (transmission plus circuit energies) in a multi-hop WSN. Each cluster is managed by one or two CHs: A *master* CH (MCH) and a *slave* CH (SCH). The two CHs operate as a cooperative multi-antenna node for inter-cluster communications (see Figure 1). The operation of CMIMO consists of three main phases: Cluster formation, intra-cluster communications, and inter-cluster communications. We discuss these phases in detail in the next subsection. In brief, cluster formation is a distributed process for selecting the CHs of each cluster (the MCH is mandatory whereas the SCH may or may not be present) and associating non-CH nodes with corresponding MCH/SCH pairs. During intra-cluster communications, the MCH is responsible for aggregating data sent by other nodes in its cluster and exchanging these data with the SCH, so that the two may operate as a cooperative multi-antenna node. Inter-cluster communications is carried out by forwarding data from the CHs of one cluster to the CHs of a neighboring cluster (or directly to the sink). An energy-efficient routing algorithm is executed over the topology of virtual nodes to determine an end-to-end inter-cluster path that minimizes the total energy consumption. This is done by running Dijkstra's algorithm with the weight of a link taken as its total energy consumption. It should be noted that for each virtual MIMO link, the MCH (or the MCH and SCH) of the receiving cluster selects the optimal transmission mode and transmission power for communication with other CHs. The mode and power can be different for different hops, depending on the distances between the CHs of neighboring clusters.

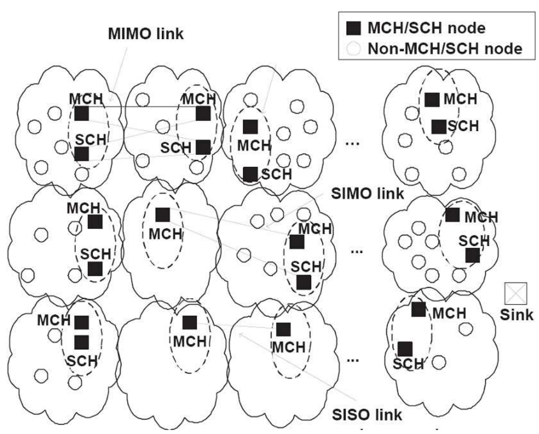

Fig. 1. Example topology of a clustered WSN with inter-cluster virtual MIMO links.

B. Operational Details

1) Cluster formation: The cluster formation process consists of the following steps:

Step 1: Neighborhood discovery. In this step, each node uses a CSMA/CA scheme to contend for the channel. Once a node v succeeds in accessing the channel, it sends a "hello" message at a power level P_{intra} to discover its 1-hop neighbors. This hello message carries the following information: node ID, its RBL (or a metric that indirectly

Step 1: Neighborhood discovery

1) Each node contends for the channel using CSMA/CA.
2) Every node sends a "hello" message with its ID, its RBL, and a list of its neighbors.
3) A node sends new "hello" messages whenever there is a change in its connectivity information.

Step 2: Selecting MCHs

1) The node that has the highest RBL in its neighborhood declares itself as an MCH.
2) ID-based criterion is used to break ties in selecting MCHs with the same RBL.
3) A node that has the highest RBL among its *undecided* neighbors becomes an MCH.
4) Any node that hears from an MCH does not compete for the role of an MCH.

Step 3: Selecting SCHs

1) Each MCH sends an "SCH invitation message" to the node whose neighbor list overlaps the most with that MCH's neighbor list.
2) The invited node waits for a duration of time (ζ) before making its decision.
3) An invited node associates itself with the closest MCH and responds with an "SCH acceptance message."
4) The MCH confirms this association via an "SCH confirmation message."
5) Non-CH nodes go to cluster membership step.

Step 4: Cluster membership

1) Every non-CH node asks its closest MCH to join its cluster via a "membership request message."
2) The MCH sends its "membership list message" periodically.
3) The "membership list message" includes the TDMA schedule that the SCH and non-CH nodes should follow in their intra-cluster data communications.

Fig. 2. Summary of the clustering procedure in CMIMO.

reflects this value), and a list of known neighbors (nodes that v has received hello messages from). Node v sends a hello message whenever any of these two events occurs: (1) v receives a hello message from a node, say u, that is not already in v's neighbor list, or (2) v receives a hello message from an already known neighbor u but u's hello message does not include v as a neighbor. In both cases, v broadcasts an updated hello message.

We explain the neighborhood discovery process through the example in Figure 3. The first number next to each node indicates the node ID, whereas the second number indicates the RBL for that node. Suppose that according to the CSMA/CA scheme, the sequence of transmissions is as follows: Nodes 1 and 5 send first (they are not neighbors, i.e., they cannot hear each other), followed by node 2, then node 4, and finally node 3. Node 1 sends its hello message, informing others about its ID (N_1), its RBL (E_1), and that it has not heard from any neighbors yet. Node 5 sends its hello message, containing its ID (N_5), its RBL (E_5), and its neighbors list, which is still empty. Node 2 sends its hello message, which contains its ID (N_2), its RBL (E_2), and its neighbors list, which has one neighbor (N_1). Node 4 sends its hello message, informing the other nodes about its ID (N_4), its RBL (E_4), and its neighbors list, which consists of N_1, N_2, and N_5. Finally, node 3 sends a hello message that includes its ID (N_3), its RBL (E_3), and its neighbor list. Suppose that the hello messages from N_1 and N_5 collided at node 3, so node 3 is aware of only two

TABLE I
FIRST ROUND OF HELLO-MESSAGE EXCHANGES.

Node ID	Heard hello messages from
N_1	–
N_5	–
N_2	N_1
N_4	N_1, N_2, N_5
N_3	N_2, N_4

neighbors (N_2 and N_4). Table I provides the neighbor lists at various nodes after the first round of hello messages.

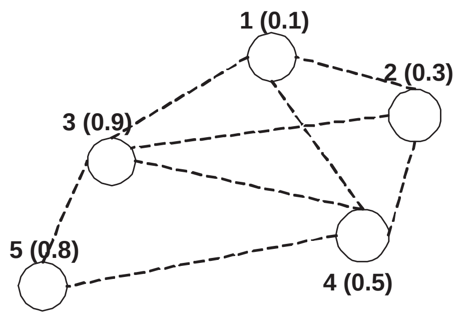

Fig. 3. Example that illustrates neighborhood discovery. Dashed lines indicate connectivity at transmission power level P_{intra}.

After the first round of hello-message exchanges, some nodes may still need to send additional hello messages if an inconsistency is detected in the neighborhood information. In our example, node 1 receives hello messages from new neighbors (N_2 and N_4) that have not been previously included in the neighbor list of node 1. Furthermore, node 1 has heard a hello message from N_3 that did not include N_1 in its neighbor list. Therefore, N_1 sends a second hello message, notifying N_2 and N_4 that it has received their hello messages, and probing N_3 to send an updated hello message that includes N_1. Node 5 has also to send a second hello message to notify N_4 that it has received its hello message, and inform N_3 that it is still waiting for N_3's updated neighbor list that includes N_5. Nodes 2, 3, and 4 need also to send additional hello messages, as they have received hello messages from new neighbors that have not been previously included in their neighbor lists. The final outcome of the second round of hello-message exchanges is given in Table II.

After sending their second hello messages, N_1 and N_5 wait for a fixed duration of time (Δ). After this Δ, if N_3 has not sent an updated list that includes N_1 and N_5, the two nodes will send duplicate hello messages (only once), following the same procedure explained above. If N_1 and N_5 do not receive a response from N_3 after sending their

TABLE II
SECOND ROUND OF HELLO-MESSAGE EXCHANGES.

Node ID	Heard hello messages from
N_1	N_2, N_3, N_4
N_5	N_3, N_4
N_2	N_1, N_3, N_4
N_4	N_1, N_2, N_3, N_5
N_3	N_1, N_2, N_4, N_5

third hello messages, they assume that N_3 has already heard their hello messages and has responded with an updated hello message, but this message collided with other packets. At this point, N_1 and N_5 terminate the neighborhood discovery process (unless new hello messages are received). Note that the neighbor list is needed only at the beginning of the (re)clustering phase. After that, a node does not need to know who its neighbors are, except for its MCH and SCH. Hence, a node does not need to estimate when a non-CH neighbor runs out of battery. The neighbor list is refreshed when a clustering request is received, which triggers a new neighborhood discovery process.

Our neighborhood discovery approach **does not implement link-layer retransmissions of hello messages**, i.e., explicit acknowledgments (ACKs) are not used. Essentially, a hello message serves as an announcement of a node's information and as an implicit ACK to other nodes' information. This reduces the protocol overhead because one hello message may replace several explicit ACKs. **By checking the broadcasted list of its neighbors, a node can infer if its recent hello message has been received by its neighbors or not. If a node does not see itself in the neighbor list of one of its neighbor, then it infers that its previous hello message was not correctly received by the neighbor. In this case, the node will resend another hello message.**

Because each node sends at most two *additional* hello messages to update and discover each of its neighbors, the neighborhood discovery process should terminate after a finite amount of time. The duration of the neighborhood discovery phase is bounded by the time (T_{upper}) it takes a node S that has the highest number of neighbors (the maximum node degree, denoted by N_{deg}) to send all of its $(1 + 2N_{deg})$ hello messages. Nodes use a CSMA/CA mechanism to access the channel and send their hello messages. Hence, the total time for a node to send a hello message, T, is comprised of backoff delay ($T_{backoff}$) before the start of a transmission, transmission delay (T_{delay}), which is the time elapsed from sending the first until the last bit of a packet, and propagation delay (T_{prog}).

Accordingly, T_{upper} is given by:

$$\begin{aligned}&T_{upper}\\&=\left(T_{backoff}+\text{DIFS}+T_{prog}+T_{delay}\right)\\&\quad+2N_{deg}\left(T_{backoff}+\text{DIFS}+T_{prog}+T_{delay}+\Delta\right)\\&=\left(2N_{deg}+1\right)\left(T_{backoff}+\text{DIFS}+T_{prog}+T_{delay}\right)+2N_{deg}\Delta\end{aligned} \tag{1}$$

where the first term in the first equation is the time to send the first hello message at node S. DIFS is the distributed inter-frame space interval, which is used right after the backoff timer reaches zero and before transmitting a hello message. Δ is the time elapsed between the second and the third hello messages.

The propagation delay is often negligible. The transmission delay depends on the message length. The backoff delay $T_{backoff}$ depends on the minimum ($W_{\min}$) and maximum ($W_{\max}$) backoff window sizes of the CSMA/CA mechanism. We assume that $W_{\max}=2^m W_{\min}$, and apply the results by Carvalho and Bianchi in [26] and [27], to obtain $T_{backoff}$:

$$T_{backoff}=\frac{\alpha\left(W_{\min}\beta-1\right)+2(1-q)(T_{prog}+T_{delay})}{2q} \tag{2}$$

where α is the average backoff step size, and β and q are given by:

$$\begin{aligned}\beta&=\frac{q-2^m(1-q)^{m+1}}{1-2\left(1-q\right)}\\q&=1-\frac{2W_{\min}N_{deg}}{\left(W_{\min}+1\right)^2+2W_{\min}N_{deg}}.\end{aligned} \tag{3}$$

Given the size of a hello message, the transmission speed of the wireless link, $W_{\min}$, $W_{\max}$, Δ (a controllable parameter), DIFS, one can compute (T_{upper}) as a function of N_{deg}. Each node independently decides to move to the next step (selecting MCHs) once no event triggers it to send more hello messages.

Step 2: Selecting MCHs. After neighborhood discovery is completed, MCHs are selected. Because MCHs do more work than non-MCH nodes (e.g., collecting, aggregating, and forwarding data), the selection criterion for MCHs is the node's RBL. It should be noted that this criterion has been also used in several previously proposed clustering protocols (e.g., [18]).

The MCH selection process can be summarized as follows. Each node maintains a table of RBL values of all its

1-hop neighbors (obtained during the neighborhood discovery process). All nodes start the clustering process in the "undecided" state. Every node compares its RBL to those of its one-hop neighbors. If the node has the highest RBL in its neighborhood, it declares itself as an MCH and announces that to its neighbors at a power level P_{intra}. Any node that receives an announcement stops competing for the role of an MCH[2]. A node that declares itself as an MCH or hears an MCH message from a neighbor switches to a "decided" state and announces this new state to its neighbors at a power level P_{intra}. The "decided" state means that the node is either an MCH or cannot be an MCH because it has already heard an MCH message. After the first round of MCH and non-MCH announcements, some nodes may still be in the "undecided" state. These nodes have not received MCH announcements, but must have received non-MCH announcements from one or more neighbors. The remaining undecided nodes repeat the above MCH selection process, but without considering non-MCH neighbors who are already in the "decided" state. The MCH selection process ensures that no two MCHs are neighbors at power level P_{intra} (which is used to transmit the hello messages).

To illustrate, consider the example in Figure 4. Initially, all nodes are "undecided." Based on their RBL values relative to their neighbors, nodes 2 and 7 declare themselves as MCHs. This forces the neighbors of 2 and 7 (including nodes 3 and 6) to switch to the "decided" state (as non-MCHs). As a result, node 4 declares itself as an MCH based on its RBL. On the other hand, N_5 has to wait for N_4 to decide because N_4 has more RBL than N_5. Since every "undecided" node continues the process until it decides to be an MCH or until it hears from an MCH, it is clear that the clustering process converges in a finite amount of time. This convergence can be proven by noting that a node has only two possibilities regarding hearing an MCH message. The node may hear an MCH message, so it changes its state to "decided," or it does not hear an MCH message. In the latter case, the neighbors of this node will either declare themselves as "decided" or remain "undecided." This forces one of them to declare itself as an MCH.

The practicality of using the RBL as a metric for selecting CHs was demonstrated in [28]. The authors formulated a simple energy model to keep track of the battery consumption of CHs and non-CH nodes. According to [28], a node can calculate the amount of energy used during its active time, and can find the percentage of its RBL.

Step 3: Selecting SCHs. The next step is to associate an SCH with each MCH, if possible. The purpose of having SCHs is to achieve VMIMO diversity gain during the inter-cluster communications phase by constructing

[2]Ties in RBL may be broken based on node ID.

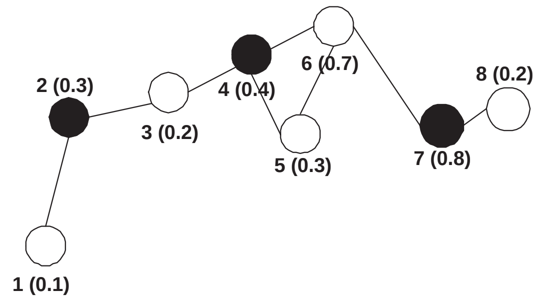

Fig. 4. Selection of MCHs based on RBL (MCHs are indicated by black circles).

cooperative multi-antenna nodes. We consider three possible criteria for SCH selection. The first one is based on the degree of overlap between the neighbor lists of the MCH and SCH. Specifically, to select SCHs, each MCH sends an "SCH invitation message" at a power level P_{intra} to a neighbor whose neighbor list overlaps the most with the MCH's neighbor list. The rationale behind this criterion is to maximize the number of nodes that will eventually be served by both the MCH and SCH. At the same time, this criterion generally favors nodes close to the MCH, hence reducing the energy needed for coordinating the cooperative MIMO operation. The second criterion for SCH selection is based on the RBL value, while the third criterion is based on proximity (signal strength). The objective of these last two criteria is to reduce energy imbalance or the energy expended for data and signalling exchange between the MCH and its SCH. Note that in CMIMO, the fading conditions at the two CHs of each cluster are very likely to be independent. Hence, CMIMO diversity gain can be realized. This follows from the fact that the probability of finding two neighboring nodes that are too close to each other (i.e., their distance is less than half of the operating wavelength) is very small.

Upon receiving the first invitation message from an MCH, the invited node waits for a fixed duration of time (ζ) before making its decision. If the invited node receives more than one invitation within ζ, it chooses the closest inviting MCH (based on the strength of the received signal). Ties in selecting SCHs can be resolved heuristically, e.g., by favoring nodes with smaller IDs. The invited node announces its decision via an "SCH acceptance message" at a power level P_{intra}. Inviting MCHs that are not accepted by the SCH can then invite other potential SCHs. Upon receiving an "SCH acceptance message," the MCH whose invitation was accepted confirms this association via an "SCH confirmation message." The purpose of this message is to inform other neighbors of this MCH that they should not expect "SCH invitation messages" from that MCH, so they can move to the next step (cluster membership). The

"SCH confirmation message" is sent at a power level P_{inter} that achieves inter-cluster network connectivity (we explain later how P_{inter} is set so as to result in a connected inter-cluster backbone of VMIMO links). The "SCH confirmation message" plays a significant role in MCH-neighborhood discovery. It includes the following fields: MCH ID, SCH ID, and a list of MCHs that the MCH has already received SCH confirmation messages from. The other steps of this discovery approach are similar to those of the neighborhood discovery explained in step 1 of this phase.

To illustrate, consider the topology in Figure 5. The outcome of steps 2 and 3 for this topology is shown in Figure 6. It should be noted that some MCHs may not have SCHs, especially if the topology is sparse. Such an MCH cannot function as a virtual multi-antenna node. It can, however, be the source node of a SISO or SIMO link, or the destination node of a SISO or MISO link.

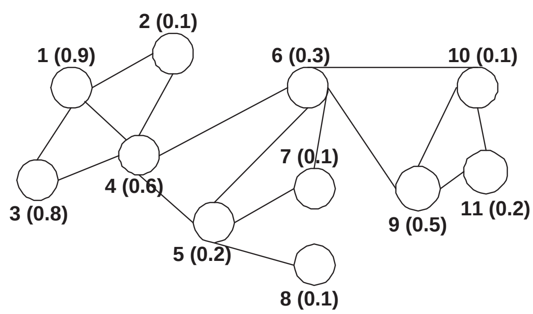

Fig. 5. Example topology that is used to illustrate the cluster-formation process.

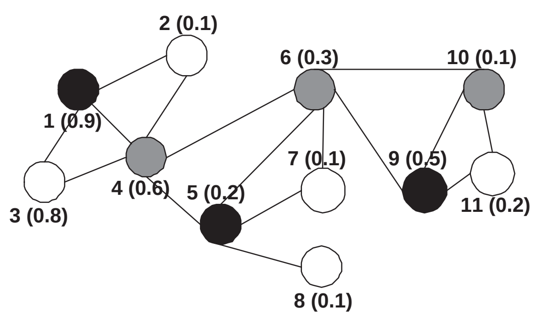

Fig. 6. MCH and SCH selection at the end of step 3 (MCHs are indicated by black circles and SCHs are indicated by grey circles).

A node v that is neither an MCH nor an SCH autonomously decides to proceed to the next step (cluster membership) once it has heard "SCH confirmation messages" from all of its MCH neighbors who are reachable from v at power P_{intra} or if no such messages are heard after a duration of time (θ).

Step 4: Cluster membership. The final step in the cluster-formation phase is to have non-CH nodes decide on which cluster to join. Because every non-CH node is a neighbor of one or more MCHs, such a node attempts to associate itself with its closest MCH by sending a "membership request message" at a power level P_{intra}.

Upon receiving a "membership request message," the MCH waits for a fixed duration of time (ϕ), allowing other non-CH nodes to send their "membership request messages" (**ϕ depends on the cluster size).** After that, the MCH sends a "membership list message" at a power level P_{intra}, announcing the IDs of the non-CH nodes that this MCH is willing to admit to its cluster. A node that does not find its ID in the "membership list message" resends (up to a maximum number of attempts) its "membership request message." It should be noted that there is some chance that the MCH will not include a given node in its updated "membership list message." This can be attributed to several reasons, e.g., the MCH wishes to limit the number of non-CH nodes in the cluster, etc. In such a case, the non-CH node tries to associate itself with the next closest MCH. Each MCH periodically announces its cluster membership list.

When an MCH sends a "membership list message," it includes the time division multiple access (TDMA) schedule that its SCH (if any) and non-CH nodes should follow in sending their data during the intra-cluster communications phase. When a new non-CH node joins a given MCH, the TDMA schedule is updated to include this node's ID, and is announced by the MCH via a "membership list message." As a convention, the SCH is the first node to transmit data according to the announced TDMA, followed by non-CH nodes. Figure 7 shows a timing diagram that illustrates this procedure for the topology of Figure 6. In this diagram, the first "membership list message" of N_5 contains one node, which is its SCH (N_6). A new node (N_7) sends a "membership request message" within the following ϕ duration, asking to join node 5's cluster. An updated "membership list message" is then sent, announcing the new list, which now contains N_6 and N_7, and so on. The result of the "cluster membership" step for the topology of Figure 5 is shown in Figure 8. A summary of the proposed clustering protocol is given in Figure 2.

Proposition 1: **The clustering phase requires $\mathcal{O}(N_{deg}T)$ amount of time (on average), where T is the time duration for a node to successfully access the channel and send a message (assuming signalling messages are of the same size). In the worst case, the clustering phase requires $\mathcal{O}(NT)$ time, where N is the network size. The total number of exchanged messages in the clustering phase is $\mathcal{O}(N)$, hence $\mathcal{O}(1)$ per node.**

Proof: **The clustering phase consists of the neighborhood discovery, MCH and SCH selection, and membership association. To complete this phase, nodes have to send hello messages (possibly multiple times), MCH or non-MCH announcement messages, SCH invitation/confirmation messages and membership request messages.**

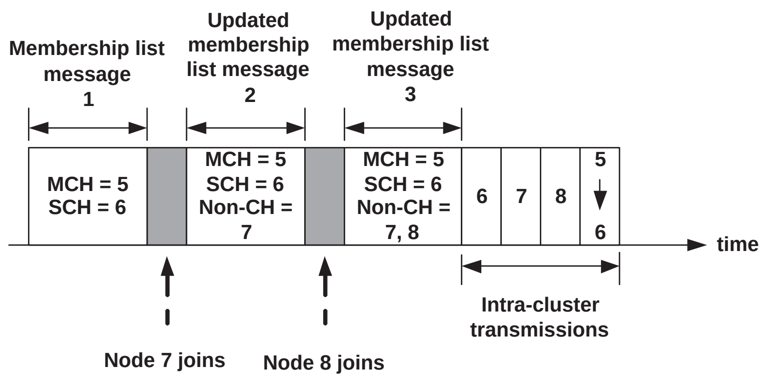

Fig. 7. Timing diagram that shows an announced TDMA schedule by MCH 5, updated forms of it, and the subsequent intra-cluster transmissions.

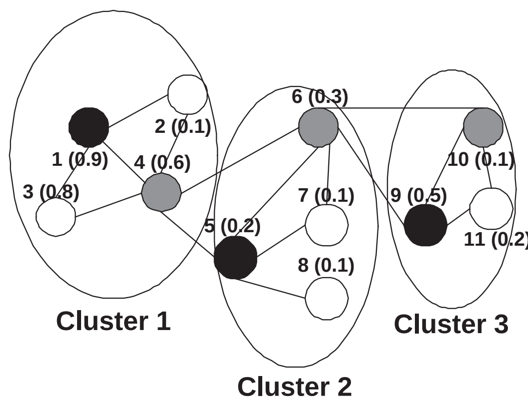

Fig. 8. WSN topology after step 4 (MCHs are indicated by black circles and SCHs are indicated by grey circles).

Hence, the total number of exchanged messages for N nodes is $\mathcal{O}(N)$. The total amount of time to send these messages by the node with the highest degree is $\mathcal{O}(N_{deg}T)$. N_{deg}'s average value is $\mu\pi R^2_{\textbf{intra}}$, leading to an average time complexity for the clustering phase of $\mathcal{O}(\mu\pi R^2_{\textbf{intra}}T)$. The worst case happens when all nodes are within the transmission range of each other (i.e., a single-hop network). In this case, the maximum node degree is $N-1$ and the total amount of time for the clustering phase is $\mathcal{O}(NT)$. □

2) Intra-cluster communications: Each non-MCH node transmits its data to the MCH at power level P_{intra} according to the TDMA schedule. After that, the node goes to sleep until its next transmission cycle. MCHs and SCHs do not go to sleep, as they may still receive data from other clusters. Upon receiving data from the SCH and non-CH nodes, an MCH aggregates the received data, and sends the aggregated data to its SCH during an assigned TDMA slot (the last slot in Figure 7). As a result, both the MCH and SCH will be ready for the inter-cluster communications phase.

Note that non-CH nodes send their data to the MCH only (and not to the SCH) because even if the cluster has an SCH, some non-CH nodes may not have the ability to directly communicate with it. For example, in Figure 8, node 8 is a non-CH node that cannot directly communicate with its SCH (node 6). However, by design all non-CH nodes must be able to directly communicate with their MCH.

3) Inter-cluster communications with virtual MIMO: We now discuss how to establish an inter-cluster VMIMO link. The purpose of this phase is to decide on the appropriate transmission power and antenna mode to use. Note that at this point, each MCH is already aware of its neighboring MCHs (at power level P_{inter}) following the overhearing of the "SCH confirmation messages" in step 3 of the cluster-formation phase. For a static WSN and a given clustering configuration, steps 1 through 6 of this phase are executed only once. These steps can be viewed as a "training process," whose outcome is the optimal VMIMO configuration for various inter-cluster links. The operational details for establishing cooperative MIMO links are as follows:

Step 1: If the MCH of a given cluster wishes to establish VMIMO links with adjacent clusters, it accesses the channel using the CSMA/CA scheme. Once it acquires the channel, the MCH broadcasts a channel-probing-request (CPREQ) packet at a power level $P^{(1)}_{CPREQ}$. One purpose of this CPREQ packet is to notify the MCH/SCH of the receiving clusters of the presence or absence of an SCH at the source cluster. The MCH's CPREQ will also be heard by the SCH of the transmitting cluster, so it knows when to send its own CPREQ.

Step 2: If the source cluster has an SCH, then after the source MCH sends its CPREQ packet, the source SCH will follow with its own CPREQ. This CPREQ is sent at $P^{(2)}_{CPREQ}$, and is used to obtain the channel state information (CSI) between the source SCH and the destination SCH and MCH. The CPREQ packets also facilitate the determination of an energy-efficient path between the source CHs and the sink, as explained later. An example that illustrates the CPREQ broadcasts between two clusters (steps 1 and 2) is shown in Figure 9.

Step 3: Upon receiving the two CPREQ packets, the receiving MCHs and SCHs estimate the CSI between the source MCH/SCH and the receiving MCH/SCH and communicate such information to each other. From that, the receiving CHs calculate the minimum power needed to communicate between the CHs of the transmitting and receiving clusters using one of four possible modes (SISO, MISO, SIMO, MIMO). Such power determination is explained in Section IV.

Step 4: The MCH and SCH in each neighboring cluster determine the optimal transmission mode that minimizes

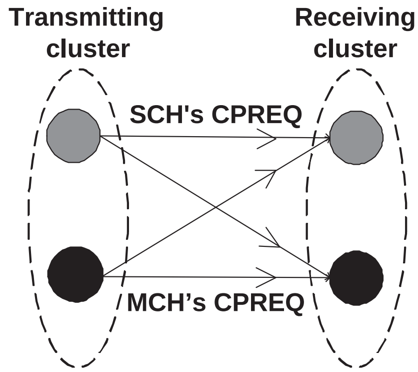

Fig. 9. Control-packet exchanges between two clusters (MCHs are indicated by black circles and SCHs are indicated by grey circles).

the total energy (which includes both transmission and circuit components) among the four modes. Each receiving MCH then sends this information back to the source MCH/SCH (and also to the receiving SCH) via a channel-probing-response (CPRES) packet. It should be noted that the process of finding the optimum transmission mode via exchanging the CPREQ and CPRES packets is done once (not on a per-packet basis), as the network is assumed to be stationary.

Step 5: A graph of virtual inter-cluster links is then constructed, where each link in this graph represents the transmission mode that requires the least amount of energy, as shown in Figure 10. Information about the selected transmission power and mode for each link is flooded throughout the network, so that each cluster will eventually have a complete knowledge of all links' weights, which will be used as input to the path selection algorithm.

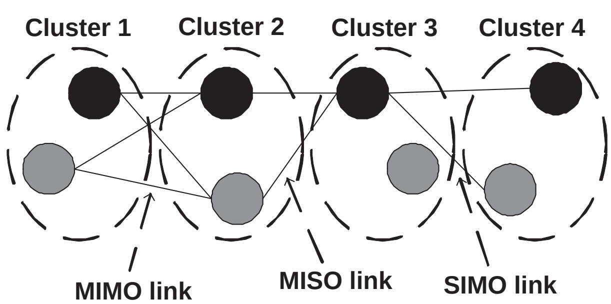

Fig. 10. Example of the WSN after step 5.

Step 6: A minimum-energy routing algorithm is then executed by the source MCH on the inter-cluster topology of virtual nodes. This algorithm consists of two steps. In the first step, all pairs of virtual nodes that can directly communicate at P_{inter} using at least one of the four modes are determined. In the second step, we run Dijkstra's algorithm with the weight of a link taken as its total energy value determined from the first step. The returned path has the minimum total energy among all possible paths between the source CHs and the sink.

Step 7: Whenever a given cluster has data to transmit to the sink, its CHs transmit their aggregated data to the CHs of the next-hop cluster using the negotiated power and mode. Specifically, the MCH of the source cluster accesses the medium using CSMA/CA and sends an RTS packet. Upon receiving the RTS, the MCH of the receiving cluster responds with a CTS packet that serves as a synchronization signal for the two CHs of the source cluster, so that they transmit their data simultaneously to achieve MIMO diversity gain.

Step 8: The MCH of the receiving cluster acknowledges the data reception via an ACK packet. If such an ACK is not received, the CHs of the transmitting cluster retransmit their data (up to a given maximum number of retransmissions).

C. Properties of CMIMO

CMIMO has several features. First, it is a distributed protocol because every node in the WSN independently makes its decisions based on local information. Second, at the end of the clustering process, a node is decided to be either a CH (master or slave) or a non-CH node. In other words, the clustering process is guaranteed to terminate. This can be easily proven by recalling that the node with the highest RBL in its neighborhood declares itself as an MCH. Such an MCH then selects an SCH according to one of the three criteria. Next, each non-CH node selects one of the MCHs to join. Third, an SCH cannot belong to more than one cluster. This is because an invited SCH responds to only one of the received requests from MCHs. Fourth, MCHs are evenly distributed because no two MCHs can be within each other's cluster range (as determined by the power level P_{intra}).

IV. Energy Model

In this section, we analyze the energy consumption in the CMIMO protocol. The purpose of this analysis is to study the tradeoff between various parameters that are used in the system design, and also to obtain the energy values of the four possible transmission modes. Following [6], the total power consumed in sending a packet consists of transmission and circuit powers. The transmission power for inter-cluster data transmissions is adjustable and is given by $P_t = (1+\delta)P_{out}$, where δ is a factor that depends on the *drain efficiency* [29] of the power amplifier and the underlying modulation scheme [30], and P_{out} is the total transmit power at the air interface. This P_{out} can be expressed as:

$$P_{out} = \gamma(M_t, M_r) N_o B N_f G_o M_l d^n \tag{4}$$

where $\gamma(M_t, M_r)$ is the SNR threshold at the receiver when M_t and M_r antennas are used for transmission and reception, respectively, N_o is the single-sided thermal noise power spectral density (PSD), B is the passband bandwidth, N_f is the receiver noise figure ($N_f \stackrel{\text{def}}{=} \frac{N_r}{N_o}$, with N_r being the PSD of the total effective noise at the receiver input), G_o is a constant that depends on the transmitter and receiver antenna gains, M_l is a link margin that compensates for hardware variations and other sources of interference, n is the path-loss exponent, and d is the transmitter-receiver distance. For a VMIMO operation, we take d to be the maximum of the four distances between the source and destination MCHs/SCHs. Note that $\gamma(M_t, M_r)$ depends on the target BER and the specific transmission mode.

The values of P_{intra} and P_{inter} are indirectly specified through the inter- and intra- cluster transmission ranges, R_{intra} and R_{inter}, respectively. In Section V-A, we derive the relation between the two so as to guarantee network connectivity. The relative ratio between P_{intra} and P_{inter} depends on the path loss factor n (n ranges from 2 to 6) and the BER requirement to successfully decode control messages. Hence, P_{intra} and P_{inter} also depend on the modulation scheme and the error correcting code.

As for the circuit power (P_c), it is given by [6]:

$$\begin{aligned} P_c \approx\ & M_t(P_{DAC} + P_{mix} + P_{filt}) + 2P_{syn} + \\ & M_r(P_{LNA} + P_{mix} + P_{IFA} + P_{filr} + P_{ADC}) \end{aligned} \tag{5}$$

where P_{DAC}, P_{mix}, P_{LNA}, P_{IFA}, P_{filt}, P_{filr}, P_{ADC}, and P_{syn} are the power consumption values for the digital-to-analog converter, the mixer, the low noise amplifier, the intermediate frequency amplifier, the active filters at the transmitter and the receiver sides, the analog-to-digital converter, and the frequency synthesizer, respectively.

Accordingly, the total energy consumption per bit is:

$$E_{bt} = \frac{P_t + P_c}{R_b} \tag{6}$$

where R_b is the bit rate. Using (4) and (5), E_{bt} can be written in terms of d, M_t, M_r, and R_b as follows:

$$E_{bt} = \frac{C_1 \gamma(M_t, M_r) d^n + C_2 M_t + C_3 M_r + C_4}{R_b} \tag{7}$$

where C_1, C_2, C_3, and C_4 are circuit-specific constants.

Note that the transmission mode (defined by M_t and M_r), $\gamma(M_t, M_r)$, and d have significant impacts on E_{bt}. For the same value of BER, the smaller the values of M_t and M_r, the larger the $\gamma(M_t, M_r)$, i.e., $\gamma(1,1) > \gamma(2,2)$, making MIMO more favorable in terms of transmission power. However, a distance-dependent tradeoff emerges between transmission and circuit powers. Whereas a 2×2 MIMO link requires less transmission power than, say, a SISO link for the same target BER, it also requires more circuit power at both ends of the link. For relatively small distances, circuit power is dominant, and hence a SISO mode is more energy-efficient than a multi-antenna mode. As the transmitter-receiver distance increases, the tradeoff shifts in favor of multi-antenna modes (SIMO, MISO, MIMO).

V. DESIGN ISSUES

A. Connectivity

We now show how CMIMO constructs a connected inter-cluster graph of MCHs.

Lemma 1: Before executing the CMIMO protocol, if the network of sensors is connected under a transmission range of R_{intra}, then after running the CMIMO protocol, the MCHs of any two neighboring nodes are either identical or are within range of $3R_{intra}$.

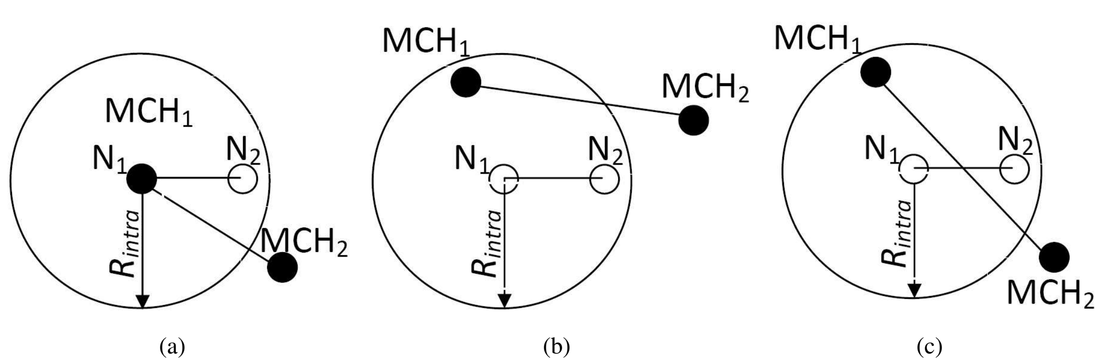

Fig. 11. MCHs of any two neighboring sensors are within range of each other if $R_{inter} = 3R_{intra}$.

Proof: The proof is illustrated in Figure 11. Because the network is connected under a transmission range of R_{intra}, every node has at least one neighbor within this range. Let node N_1 be a neighbor of node N_2. After running the CMIMO protocol, N_1 and N_2 may belong to one cluster (they have the same MCH) or to two different clusters. In the later case, both N_1 and N_2 cannot be MCHs at the same time (because no two MCHs can be within the intra-cluster range) or none of the two are MCHs . If N_1 is an MCH (Figure 11(a)), it is easy to show using the triangle inequality that the distance between the two MCHs is less than $2R_{intra}$. For the case when both N_1 and N_2 are non-MCHs (Figure 11(b) and 11(c)), we apply the triangle inequality twice to show that the distance between the two MCHs is less than $3R_{intra}$. Hence, the lemma is proven.

The following theorem claims the connectivity of the graph of MCHs.

Theorem 1: The graph of MCHs generated by the CMIMO protocol is connected if $R_{inter} = 3R_{intra}$.

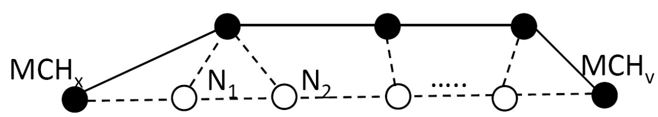

Fig. 12. A path consisting of only MCHs from an MCH to another can always be constructed when $R_{inter} = 3R_{intra}$.

Proof: Consider the subgraph $\acute{G}$ of MCH nodes. According to the CMIMO protocol, two MCHs on $\acute{G}$ are directly connected if their distance is less than R_{inter}. We will show that if $R_{inter} = 3R_{intra}$, then there exists a path on $\acute{G}$ between any two MCH nodes. Consider two such nodes, denoted by MCH_x and MCH_v (see Figure 12). It is assumed that before executing CMIMO, the network graph G (which includes all nodes) is connected at transmission range R_{intra}. Thus, there exists a path $Q = (\text{MCH}_x, N_1, N_2, \ldots, N_Q, \text{MCH}_v)$ on G that connects MCH_x and MCH_v using the transmission range R_{intra}. We call Q the primary path. Suppose that MCH_x and MCH_v are not neighbors on path Q (otherwise, the assertion is proved). Consider the next two nodes on Q, N_1 and N_2. After running the CMIMO protocol, if these two nodes belong to the same cluster, then we can replace their direct link on Q by their common MCH (see the figure). According to Lemma 1, this common MCH is itself MCH_x or is at most $3R_{intra}$ away from MCH_x. Because $R_{inter} = 3R_{intra}$, this common MCH has a direct link to MCH_x at transmission range R_{inter}. In the case that N_1 and N_2 belong to different clusters, then their corresponding MCHs are separated by no more than $3R_{intra}$, so they are directly connected on $\acute{G}$ when the transmission range is $3R_{intra}$. The process of replacing the links on Q by direct links on $\acute{G}$ can be repeated for every link on Q, leading to a path between MCH_x and MCH_v on $\acute{G}$.

We now compare the connectivity condition in CMIMO with that of a previously proposed clustering scheme, namely HEED [18], as the system model in HEED is close to that of CMIMO, but with one CH per cluster. The authors of [18] found that any two CHs can communicate if $R_{inter} \geq 6R_{intra}$ so that the network is connected. In contrast, the requirement in Theorem 1 gives higher chances for inter-cluster communications than HEED, as a smaller range is reserved for inter-cluster communications (i.e., more spatial reuse).

B. Synchronization

One of the important design issues in operating CMIMO is synchronization. The importance of this aspect comes from the fact that synchronization allows CHs to fully exploit MIMO gains for inter-cluster communications. Extensive research has been done on quantifying the tradeoff between implementing synchronous network operation and the overhead and inaccuracy associated with such operation.

Several approaches were proposed to solve the synchronization issue. One of them is the reference-broadcast synchronization (RBS) technique [31], which we can adopt for CMIMO, as follows. An MCH asks one of its cluster nodes (other than the SCH) to send RBS beacons. After exchanging the RBS beacons, the MCH and SCH start sending the data simultaneously to the CHs of a receiving cluster. Another approach relies on the knowledge of the transmission delay and channel estimation [9]. The drawbacks of these approaches include the large number of generated messages, the long time to achieve synchronization, and the lack of consideration of the energy requirements of various nodes.

To address these drawbacks, we propose another solution to the synchronization issue. The key idea is to select a node in the network to act as a beacon cluster head (BCH). Such a node sends beacons to neighboring MCHs so that they can adjust the start times of their frames accordingly. The BCH must be one of the MCHs to enable it to communicate with neighboring MCHs. To satisfy this condition, one node is arbitrarily chosen during the design phase (before starting the "cluster formation" phase) to be a BCH. After MCHs are selected, if the BCH turns out to be a non-MCH node (SCH or non-CH node), it asks its corresponding MCH to take its role as a BCH according to the following approach. If the selected BCH happens to be a non-CH node, it notifies its MCH to take its role via the "membership request message," which is acknowledged by the MCH through the "membership list message." Otherwise, if the BCH happens to be an SCH, it asks its MCH to take its role via the "SCH acceptance message," which is acknowledged by the MCH via the "SCH confirmation message."

According to the above approach, only one node in the network acts as a BCH. Therefore, all other nodes know that they should hear beacons from the BCH. The beacons are sent by the BCH at a power level P_{inter} that satisfies the connectivity criterion. The beacon is sent at the beginning of the frame, so that it represents a reference to the neighboring MCHs. Any MCH that hears the beacon adjusts the beginning of its frame accordingly, and retransmits the beacon to its neighboring MCHs. Any MCH that hears a beacon does not relay this beacon until it adjusts its frame first. According to this proposed approach, all MCHs will have synchronous frames. Note that if the RBL of a BCH falls below a given threshold, the BCH resigns from its role and sends a reclustering request.

C. Reclustering

The key idea for reclustering is that when the RBL of any MCH falls below a specific threshold (e.g., 20% of its initial value), this MCH sends a "reclustering" message to its neighboring MCHs at power level P_{inter}. This message will be heard by the requesting MCH's non-CH nodes, its SCH, and its neighboring MCHs. Reclustering messages are similar to the "hello" messages used in the neighborhood discovery process in the cluster-formation phase, except that they are sent at a higher power level (P_{inter}). **Neighboring MCHs that hear a reclustering message relinquish their cluster head role and invoke a neighborhood discovery process**. The rationale behind restricting the reclustering request to MCHs is that in most cases, MCHs are the ones that deplete their batteries first (before SCHs and non-CH nodes) because of their additional responsibilities in aggregating data, sending it to the SCH, and forwarding it to CHs in neighboring clusters. Reclustering may also be performed as a result of topological changes, but this aspect is not considered here because the network is assumed to be stationary.

D. Medium Access Control

We now discuss an issue related to the MAC layer, namely how to ensure reliable communications (i.e., taking packet losses into account). It should be emphasized that our work is mainly focused on the clustering/routing aspects of CMIMO, which take place at layers above the MAC layer.

Reliable communication is established by using implicit and explicit ACKs at the MAC layer for all network transmissions. These include transmissions for neighborhood discovery (where hello messages represent implicit ACKs of previously heard hello messages), between a non-CH node and its MCH (e.g., a "membership request message" is acknowledged by a "membership list message"), between an SCH and its MCH (e.g., an "SCH acceptance message" is acknowledged by an "SCH confirmation message"), and between an SCH/MCH in a transmitting cluster and an SCH/MCH in a receiving cluster (e.g., MCH's and SCH's CPREQ packets are acknowledged by MCH's CPRES). In addition, CMIMO retransmits the packets that have not been acknowledged (up to a certain limit on retransmissions) to increase the chances of correctly receiving these packets.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of CMIMO via simulations. We also compare it with the distributed clustering algorithm (DCA) [17], which resembles CMIMO in the criterion used to select CHs. The primary goal of this comparison is to demonstrate the benefits of cooperative MIMO over a single-antenna system (represented by DCA). Recall that CMIMO adapts the transmission mode and power on a per-packet basis. On the other hand, all transmissions in DCA take place using the SISO mode, where each cluster has only one CH.

We consider 100 stationary nodes that are randomly deployed in a square of length $L_{\max}$. Unless stated otherwise, we take $L_{\max} = 1000$ meters, $R_{intra} = 250$ meters, and $R_{inter} = 750$ meters. **Operating frequency is 2.5 GHz. In this case, the probability of finding two neighboring nodes whose distance is less than half wavelength is about 1.3×10^{-4}.** The sink is located to the right of the square, at a horizontal distance of 333 meters and a vertical distance of 500 meters from the bottom right corner of the field. Each node generates packets at rate λ packets per second. The sink is the only node that is physically equipped with two antennas, enabling it to operate as a real multi-antenna node. Multi-hop operation based on minimum-energy routing is used for inter-cluster communications. The values of $\gamma(M_t, M_r)$ that are required to achieve a target BER of 0.001 are taken from [6] to be: 24.4 dB for SISO, 10.6 dB for SIMO, 14.1 dB for MISO, and 6.9 dB for MIMO. For the wireless channel, we assume a Rayleigh fading model along with a distance-dependent path loss, which has a power falloff of d^4. Other simulation parameters are given in Table III. Our results are based on simulation experiments conducted using CSIM (a C-based process-oriented discrete-event simulation package [32]). **Each plot represents the average of 10 simulation runs.**

Data-packet size	2000 bytes
Control-packet size	20 bytes
P_{DAC}, P_{ADC}	15 mW
P_{mix}	30.3 mW
P_{filt}, P_{filr}	2.5 mW
P_{syn}	50 mW
P_{LNA}	20 mW
P_{IFA}	2 mW
M_l, N_f	10 dB

TABLE III
SIMULATION PARAMETERS

A. *Energy Consumption*

Figure 13 shows the impact of R_{inter} on the total energy consumption for CMIMO and DCA. In this experiment, SCH selection is done based on the amount of overlap between the neighbor lists of the MCH and SCH. The results illustrate that energy consumption increases with R_{inter}. CMIMO significantly outperforms DCA in energy consumption. The superiority of CMIMO over DCA becomes more apparent when R_{inter} is large. This is attributed to the fact that the higher the inter-cluster range, the higher the number of neighboring MCHs of a given MCH, and hence the higher the tendency to use multi-antenna modes for inter-cluster communications (transmission energy dominates the circuit energy in this regime, making MIMO/MISO/SIMO more energy-efficient than the SISO mode).

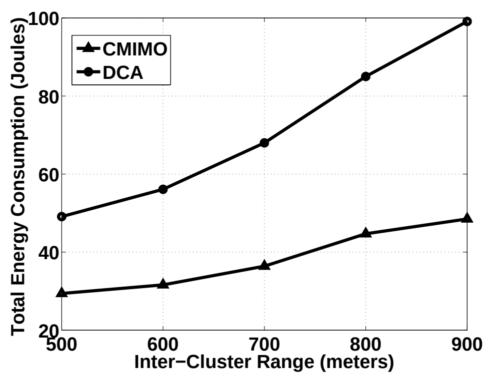

Fig. 13. Total energy consumption versus R_{inter}.

Figure 14 depicts the energy consumption versus $L_{\max}$. Because the number of nodes is kept constant (100) and, increasing $L_{\max}$ results in better energy performance for CMIMO compared with DCA. This is attributed to the fact that for large networks, the transmission energy dominates the circuit energy, forcing CMIMO to make more frequent use of MIMO/SIMO/MISO modes.

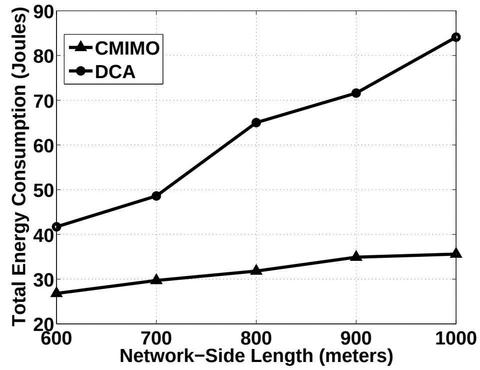

Fig. 14. Total energy consumption versus network size.

The impact of node density on the energy consumption is shown in Figure 15, which indicates that the total energy consumption increases with the number of nodes. It should be noted that CMIMO reduces the total energy consumption compared with DCA. This reduction becomes more noticeable under small number of nodes. The reason is that the distances between clusters become larger as the number of nodes decreases. In such a case, SISO becomes less preferable than other modes, which makes CMIMO more energy-efficient than DCA.

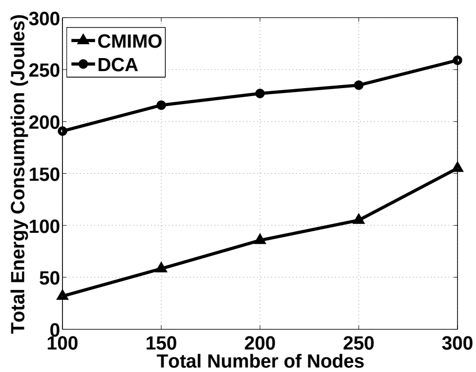

Fig. 15. Total energy consumption versus number of nodes.

B. Transmission Modes

Figure 16 depicts the percentage of time each transmission mode is used as a function of R_{inter}. For a single VMIMO link, it is known that when both transmission and circuit energies are account for, there exists a critical distance (D^*) above which VMIMO communications is more energy-efficient than SISO [6]. When $R_{inter} > D^*$, RF transmission energy dominates the overall consumed energy, so using more cooperating antennas leads to lower overall energy consumption. In this regime, our CMIMO protocol favors the 2×2 MIMO mode over other modes (see, for example, the case when $R_{inter} = 900$ meters). On the other hand, when $R_{inter} < D^*$, circuit energy is dominant, and the protocol favors SISO because of its lower circuit energy consumption. Note that R_{inter} is the *average* inter-cluster range. Variations around this average result in variations in the optimal mode for the same average value of R_{inter}.

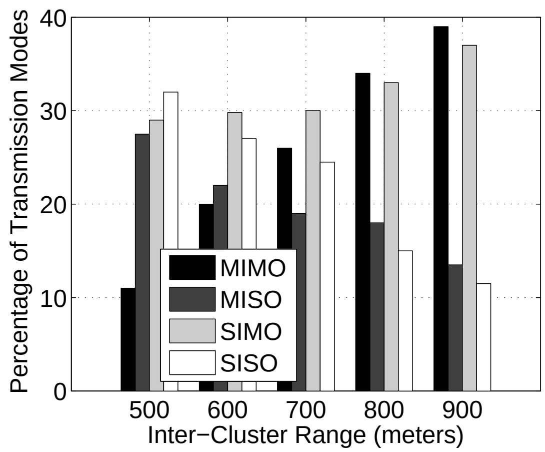

Fig. 16. Percentage of transmission modes versus inter-cluster range.

C. Number of Hops

We now study the impact of inter-cluster range and network area on the number of hops used for inter-cluster communications to the sink. Figure 17 shows the histogram of the number of hops needed for routing under different inter-cluster ranges. As expected, under small inter-cluster ranges (e.g., 300 meters), routes with large numbers of hops exist (10 hops). On the other hand, when the inter-cluster range becomes large (e.g., 900 meters), fewer hops are needed (3 hops).

The impact of the network area on the number of hops is shown in Figure 18. The figure reveals that large network areas require more hops between the communicating CHs and the sink. For example, a path with 15 hops exists when $L_{\max} = 1000$ meter. However, no more than 8 hops are needed when $L_{\max} = 600$.

D. Energy Consumption of Control Packets

We now study the energy overhead of control packets for CMIMO and DCA. Figure 19 depicts the ratio of consumed energy of control and data packets versus the data packet size. The control overhead of CMIMO becomes similar to that of DCA as the data packet size increases.

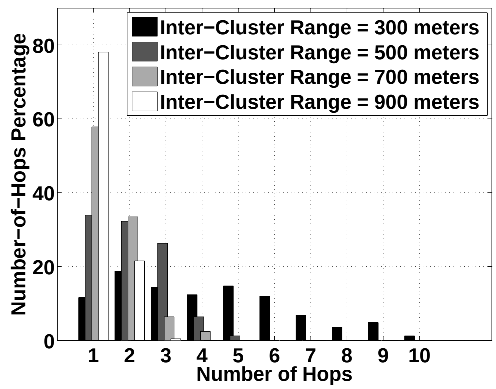

Fig. 17. Histogram of inter-cluster hop count under different inter-cluster ranges.

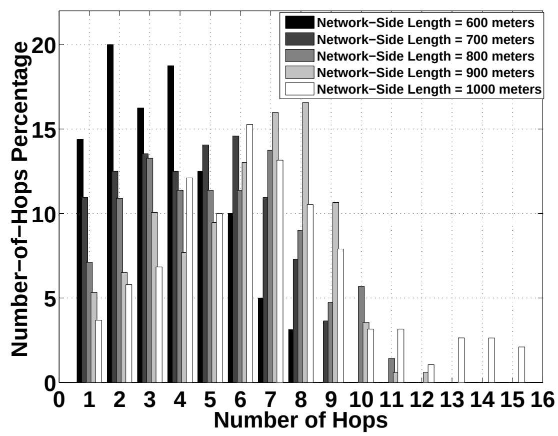

Fig. 18. Histogram of inter-cluster hop count under different network areas.

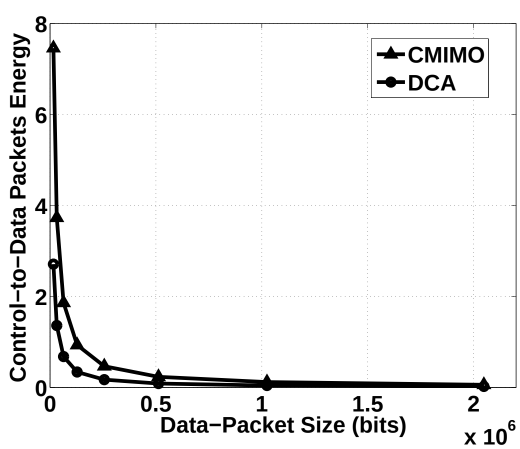

Fig. 19. Percentage of control-to-data packets energy versus data-packet size.

R_{inter} (m)	Average # of clusters (also MCHs for CMIMO)	Average # of SCHs (CMIMO)	Average # of non-CH nodes	
			CMIMO	DCA
200	16	15.8	68.2	84
300	9	9	82	91
400	5.5	5.5	89	94.5
500	4.5	4.5	91	95.5
600	4	4	92	96

TABLE IV
EFFECT OF INTER-CLUSTER RANGE ON CLUSTERING STATISTICS

Number of nodes	Average # of clusters (also MCHs for CMIMO)	Average # of SCHs (CMIMO)	Average # of non-CH nodes	
			CMIMO	DCA
50	11.75	11.25	27	38.25
100	14.33	14.33	71.33	85.67
150	15	15	120	135
200	15.25	15.25	169.5	184.75
250	15.5	15.5	219	234.5

TABLE V
EFFECT OF NUMBER OF NODES ON CLUSTERING STATISTICS

E. Clustering Statistics

Table IV depicts the effect of the inter-cluster range (R_{inter}) on clustering statistics for CMIMO and DCA. For both protocols, the number of clusters in the network decreases with R_{inter}, as expected. The number of MCHs in CMIMO is similar to the number of clusters, as each cluster must have one MCH. However, not every cluster has an SCH (although in our experiments, almost all clusters had SCHs). As expected, the number of non-CH nodes increases with R_{inter}. This is due to the fact that fewer clusters are formed in the network, which results in more non-CH nodes per cluster.

The other factor that affects the clustering statistics is the number of nodes in the network, which is studied in Table V. The table reveals that the numbers of clusters, MCHs, SCHs, and non-CH nodes increase with the number of nodes for both CMIMO and DCA.

Figures 20 compares the network lifetime under DCA and CMIMO protocols using two different network lifetime definitions: time until the network is disconnected and the time until the death of the first node. The CMIMO protocol doubles the network lifetime relative to the DCA protocol. The network lifetime improvement under CMIMO becomes more significant at a higher traffic intensity (the probability that a node has data to send in a given time slot). This

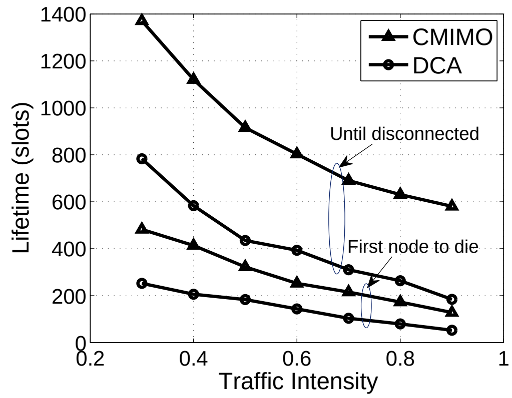

Fig. 20. Network lifetime vs. traffic intensity.

is because a higher number of packets better amortizes the cooperation overhead of CMIMO. The network lifetime decreases with the traffic intensity. From this figure, one can observe that the lifetime under the first-node-to-die definition is more sensitive (steeper slope) to traffic intensity. This is due to the fact that with a higher number of packets, the probability of a first-node-to-die event increases.

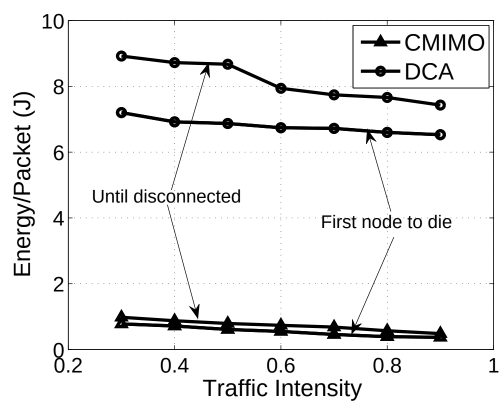

Fig. 21. Average energy per packet vs. traffic intensity.

The average energy per received packet at the sink (averaged over all packets during the network lifetime) is plotted in Figure 21. By leveraging the diversity gain, energy consumption per packet under CMIMO is much lower than that under the DCA protocol. For both DCA and CMIMO, the energy per packet with the disconnectivity-based network lifetime definition is slightly higher than that under the first-node-to-die definition. This is because in the former definition, the network continues to operate after several node failures, which may break energy-efficient paths to the sink and force packets to traverse less energy-efficient paths.

We also investigate the effectiveness of the three criteria for selecting SCHs. Figure 22 compares the network lifetime (using the first-node-to-die definition) under these criteria. Network lifetime under the first criterion (SCH is the one with highest RBL) is the longest among the three; about 14% and 30% longer than that of the second and third

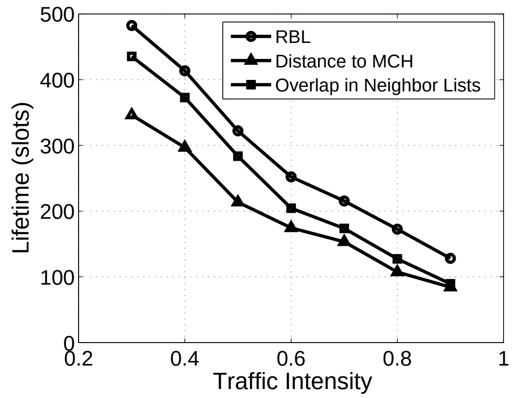

Fig. 22. Network lifetime under CMIMO with different SCH selection criteria.

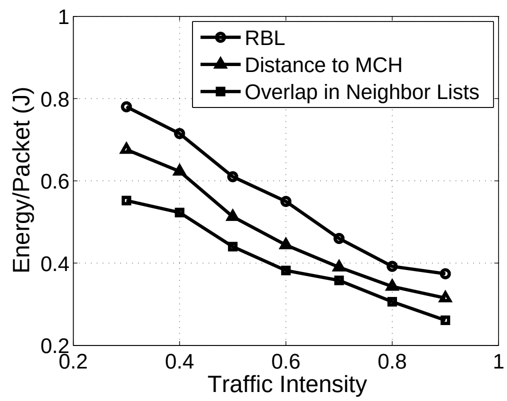

Fig. 23. Average energy per packet under CMIMO with different SCH selection criteria.

selection criteria, respectively. Figure 23 depicts the average energy per packet for the three SCH selection criteria. Although the first criterion yields the longest lifetime, its energy per packet is also the highest. This is because this criterion aims at alleviating the energy imbalance rather than improving the energy efficiency. Selecting the SCH to be the node that has the highest number of common neighbors with the MCH results in the least energy per packet.

VII. CONCLUSIONS AND FUTURE EXTENSIONS

We proposed a distributed and adaptive clustering/routing protocol (CMIMO) to minimize the total energy consumption for a multi-hop WSN. CMIMO **partitions a WSN into a number of clusters** that have one or two CHs per cluster. In addition to routing and data aggregation/fusion functions, these CHs are also responsible for inter-cluster communications. The proposed CMIMO protocol has the ability to adapt the transmission mode (SISO, MISO, SIMO, MIMO) and transmission power on a per-packet basis for inter-cluster transmissions. We studied the performance of CMIMO via simulations. The results indicate that the proposed protocol achieves significant reduction in the overall energy consumption compared with non-adaptive schemes that are designed with one CH per cluster and that use SISO (e.g., WSNs operated with the DCA scheme). **Consequently, the network lifetime under**

CMIMO is significantly improved compared with SISO-based protocols. The significance of CMIMO becomes more noticeable under large inter-cluster ranges and network sizes. **We also derived the conditions on the inter- and intra- cluster communication ranges to guarantee network connectivity. These conditions are tighter than those previously reported in the literature, and are applicable to any clustering protocol.**

One possible extension to our work is to study the impact of the modulation order (b) on the total energy consumption **and network lifetime**. b represents the number of bits per symbol, which varies with the modulation scheme. A higher value of b necessitates a higher $\gamma(M_t, M_r)$, i.e., more transmission power is required, as depicted in Figure 24 [6]. However, it also means a higher transmission rate, which in turn reduces the energy consumption. The confluence of the two effects determines the optimal modulation order (b^*). As shown in Figure 25, b^* generally decreases with the transmitter-receiver distance, as transmission energy becomes more dominant.

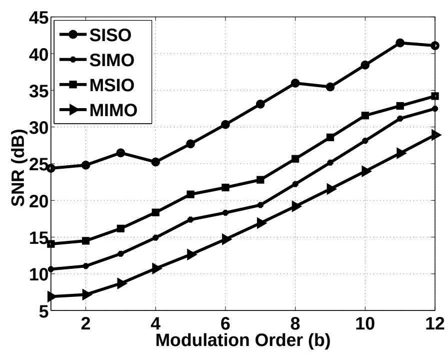

Fig. 24. $\gamma(M_t, M_r)$ versus modulation order for different transmission modes (obtained from [6]).

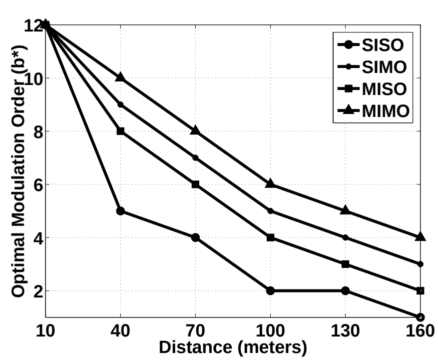

Fig. 25. b^* vs. transmitter-receiver distance for various antenna configurations.

Our work considered only MIMO's diversity gain. An interesting future extension is to incorporate other MIMO gains in the same optimization framework, i.e., allow the protocol to dynamically switch between different types of

MIMO gains (and antenna modes), depending on both energy consumption and throughput considerations.

References

[1] M. Z. Siam, M. Krunz, and O. Younis, "Energy-efficient clustering/routing for cooperative MIMO operation in sensor networks," in *Proceedings of the IEEE INFOCOM Conference*, Apr. 2009.

[2] A. Paulraj, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. Cambridge University Press, Cambridge, U.K., 2003.

[3] L. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *Proceedings of the International Conference on Distributed Computing Systems*, 2002, pp. 575–578.

[4] Y. Chen, A. Liestman, and J. Liu, "Energy-efficient data aggregation hierarchy for wireless sensor networks," in *Proceedings of the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks*, Aug. 2005.

[5] *http://www.ee.ucla.edu/∼babak/research.html*.

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

[7] M. Dohler, E. Lefranc, and H.Aghvami, "Virtual antenna arrays for future wireless mobile communication systems," in *Proceedings of the ICT Conference, Beijing, China*, June 2002.

[8] ——, "Space time block codes for virtual antenna arrays," in *Proceedings of the PIMRC Conference, Lisbon, Portugal*, September 2002.

[9] X. Li, "Energy efficient wireless sensor networks with transmission diversity," *IEE Electronics Letters*, vol. 39, no. 24, pp. 1753–1755, Nov. 2003.

[10] S. K. Jayaweera, "Energy analysis of MIMO techniques in wireless sensor networks," in *Proceedings of the 38th Annual Conference on Information Sciences and Systems (CISS)*, Mar. 2004.

[11] Q. Gao, Y. Zuo, J. Zhang, and X.-H. Peng, "Improving energy efficiency in a wireless sensor network by combining cooperative MIMO with data aggregation," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 3956–3965, 2010.

[12] W. Jaafar, W. Ajib, and D. Haccoun, "On the performance of distributed-STBC in multi-hop wireless relay networks," in *Proceedings of the European Wireless Conference*, 2010, pp. 223–230.

[13] S. Jayaweera, "Virtual MIMO-based cooperative communication for energy-constrained wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 5, pp. 984–989, 2006.

[14] Q. Qu, L. B. Milstein, and D. R. Vaman, "Cooperative and constrained MIMO communications in wireless ad hoc/sensor networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 10, pp. 3120–3129, Oct. 2010.

[15] Z. Xu, C. Long, C. Chen, and X. Guan, "Hybrid clustering and routing strategy with low overhead for wireless sensor networks," in *Proceedings of the 2010 IEEE International Conference on Communications*, May 2010, pp. 1–5.

[16] O. Younis, M. Krunz, and S. Ramasubramanian, "Node clustering in wireless sensor networks: Recent developments and deployment challenges," *IEEE Network - Special Issue on Wireless Sensor Networking*, vol. 20, no. 3, pp. 20–25, May 2006.

[17] S. Basagni, "Distributed clustering for ad hoc networks," in *Proceedings of the Fourth International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN)*, June 1999, pp. 310–315.

[18] O. Younis and S. Fahmy, "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366–379, Oct.-Dec. 2004.

[19] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[20] A. Dabirmoghaddam, M. Ghaderi, and C. Williamson, "Energy-efficient clustering in wireless sensor networks with spatially correlated data," in *Proceedings of the IEEE INFOCOM Conference*, Mar. 2010, pp. 1–2.

[21] M. Joshi, L. Osborne, B. Sun, and S. Makki, "Hot spot aware energy efficient clustering approach for wireless sensor networks," in *Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC)*, Jan. 2011, pp. 585–589.

[22] A. Youssef and M. Younis, "Overlapping multihop clustering for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 12, pp. 1844–1856, Dec. 2009.

[23] A. Chamam and S. Pierre, "On the planning of wireless sensor networks: Energy-efficient clustering under the joint routing and coverage constraint," *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1077–1086, Aug. 2009.

[24] 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*, vol. 2006, no. 72493, pp. 1–9, May 2006.

[25] D. Gong, M. Zhao, and Y. Yang, "A multi-channel cooperative MIMO MAC protocol for wireless sensor networks," in *Proceedings of the 7th IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, Nov. 2010, pp. 11–20.

[26] M. Carvalho and J. Garcia-Luna-Aceves, "Delay analysis of ieee 802.11 in single-hop networks," in *Proceedings of the 11th IEEE International Conference on Network Protocols*, 2003, pp. 146–155.

[27] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[28] O. Younis and S. Fahmy, "An experimental study of routing and data aggregation in sensor networks," in *Proceedings of the IEEE International Workshop on Localized Communication and Topology Protocols for Ad Hoc Networks (LOCAN)*, Nov. 2005.

[29] T. H. Lee, *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge University Press, Cambridge, U.K., 1998.

[30] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-constrained modulation optimization," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2349–2360, Sept. 2005.

[31] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI)*, Dec. 2002, pp. 147–163.

[32] *http://www.mesquite.com*, Mesquite Software Incorporation.