

# Adaptive Cluster-Based Data Collection in Sensor Networks with Direct Sink Access

Mahdi Lotfinezhad, *Student Member, IEEE*, Ben Liang, *Senior Member, IEEE*, and Elvino S. Sousa, *Senior Member, IEEE*

**Abstract**—Recently wireless sensor networks featuring direct sink access have been studied as an efficient architecture to gather and process data for numerous applications. In this paper, we focus on the joint effect of clustering and data correlation on the performance of such networks. More specifically, we propose a novel Cluster-based Data Collection scheme for sensor networks with Direct Sink Access (CDC-DSA), and provide an analytical framework to evaluate its performance in terms of energy consumption, latency, and robustness. In our scheme, CHs use a low-overhead and simple medium access control (MAC) conceptually similar to ALOHA to contend for the reachback channel to the data sink. Since in our model data is collected periodically, the packet arrival is not modeled by a continuous random process and, therefore, we base our framework on a transient analysis rather than a steady state analysis. Using random geometry tools, we study how the optimal average cluster size and energy savings, under the proposed MAC, vary according to the level of data correlation in the network. Extensive simulations for various protocol parameters show that our analysis is fairly accurate for a wide range of parameters. Our results suggest that despite the tradeoff between energy consumption and latency, both of which can be substantially reduced by a proper clustering design.

**Index Terms**—Wireless sensor networks, adaptive clustering, data collection latency, energy efficient communication, data correlation.

## I. INTRODUCTION

Inspired by ever-growing interesting sensor applications, numerous architectures have been proposed and extensively investigated for large scale energy-limited Wireless Sensor Networks (WSNs) [1][2][3] [4][5][6][7]. In this paper, we are interested in the following scenarios:

- A network that measures the current chemical or nuclear levels, or environmental parameters in a deserted area or an ocean, and reports the measured data to a mobile node that can be an airplane, a helicopter, or an LEO satellite.
- A network that upon the request by the data sink, e.g., an airplane, becomes active to determine whether an intruder or a set of intruders, e.g., animals, are entering or leaving a given monitored area.
- A network in a battle-field that collects the enemy-related data and reports it to a flying airplane. Note that in a battle-field sensors may not be allowed to constantly communicate with each other for security purposes. Therefore, it is very important that they start communication only when it is required. Clearly, data

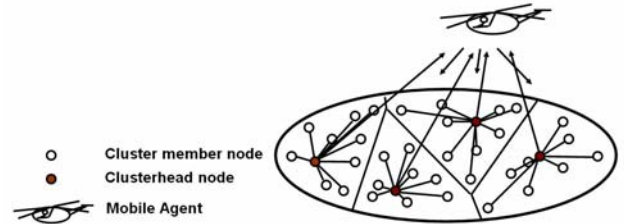


Fig. 1. Architecture of a typical network generated by CDC-DSA.

collection and forwarding should be performed with the least possible delay.

The above scenarios show that there are practical situations where propagation channels are likely line-of-sight, and hence, direct communication between sensors and the sink is possible over long distances [8][9][10]. This property allows us to avoid much of the overhead associated with medium access control and routing [10]. In other words, direct sink access significantly reduces the complexity of creating and maintaining a backbone to forward data towards the data sink.

Closely related to the architecture considered in this paper, the authors of [11] have proposed sensor networks with mobile access points (SENMA) in which mobile sinks can be manned/unmanned aerial or ground vehicles with power generators<sup>1</sup>. Mobile sinks, and generally, data sinks, in contrast to ordinary sensors, are powerful hardware units with sophisticated transceivers. For instance, they may have multi packet reception capability (MPR) [12]. When required by the application or periodically, mobile sinks visit the network to gather data or perform network maintenance. It has been shown that the simple topology of SENMA leads to significant energy savings, and improves the scalability of sensor networks [11][10].

Sink mobility in many scenarios allows direct communication between sensors and the sink, opening up many interesting problems in MAC [13][8][10]. The past literature, however, has rarely focused on the effect of clustering in SENMA and sensor networks with direct sink access [9]. In this paper, we propose and analyze a cluster-based data collection scheme for sensor networks with direct sink access (CDC-DSA) whereby stationary sensors directly communicate

*The authors are with the Department of Electrical and Computer Engineering, University of Toronto, Ontario, Canada, M5S 3G4. E-mail: {mlotfinezhad, liang}@comm.utoronto.ca, es.sousa@utoronto.ca.*

<sup>1</sup>For the rest of the paper, the terms sink and access point are used interchangeably.

with clusterheads (CHs), and therefore, form one-hop clusters. CDC-DSA generates clusters periodically in a random manner using any general clustering algorithm, for example the one used in [3]. In each sink-initiated round of data collection, CHs based on TDMA receive data samples from their cluster-members, and after data aggregation, they contend to access the reachback channel to transmit the aggregated data to the data sink. Fig. 1 shows one snapshot of a WSN using CDC-DSA.

Thanks to the simple topology generated by CDC-DSA, CHs can use a simple MAC similar to ALOHA, to contend for the reachback channel. The key point in our model is that CHs have only one packet (possibly with a different size compared to the size of raw packets generated by sensors) and after a successful transmission, they will go to the sleep mode until the next round of data collection. Therefore, the number of contending CHs decreases with time, implying that our MAC must focus on transient analysis. This is one aspect of our model which distinguishes our MAC analysis from other classical ones. CDC-DSA tunes the probability of transmission in each time slot in order to best take advantage of the MPR capability of the sink, and minimize the latency of data collection. We show that the optimal transmission probabilities can be obtained by a dynamic programming approach, and propose a simpler alternative solution. We further prove that the alternative solution is asymptotically optimal, and show that even for small networks the alternate approach performs almost optimally. In addition, we study the robustness of CDC-DSA when only partial information is known to the protocol.

In our framework, we adopt a general correlation model [14] in order to investigate the effect of correlation on both the energy consumption and data collection latency. We show that in typical scenarios, even for low correlation levels, substantial energy savings can be achieved. Furthermore, we study the effect of various system parameters on the optimal CH probability, or equivalently the optimal average cluster size, minimizing energy consumption.

In contrast to previous works on clustering algorithms, which assume a contention and error free MAC and/or do not consider a general correlation model [15][16][5], our cross-layer adaptive approach takes into account the effects of MAC and correlation on the clustering performance. To the best of our knowledge, this is the first work in the area of WSN's with direct sink access that analytically investigates the performance of cluster-based data collection under realistic MAC assumptions and correlation models. It is important to note that our results in this paper do not necessarily require the mobility of the data sink, and hold as long as sensors can directly communicate with the data sink.

The rest of this paper is organized as follows. In Section II, we describe the related work in detail. In Section III, we explain the underlying reachback channel characteristics and the correlation model. Details about our network architecture and protocol are provided in Section IV. In Section V, we provide our analytical framework to evaluate the performance of the proposed protocol. We then present our numerical and simulation results in Section VI. Finally, we conclude the paper in Section VII.

## II. RELATED WORK

Perhaps the most immediate sample of WSN's with direct sink access is a WSN with a mobile data sink. The idea of introducing mobile nodes has been mainly inspired by the fact that node's mobility saves energy by allowing short-hop communication instead of long multi-hop communication [17][18][19][11]. On the other hand, some researchers have proposed to exploit the node's purposeful mobility to ensure sufficient coverage [20]. Two main categories have been developed for the research on mobile sinks. In the first category, the mobile sink still uses multi-hop routes to access sensors' data. Moreover, the coverage area of the mobile sink is comparable with the one for ordinary sensors, i.e., they are short range devices [18][19][21]. For instance, in [19], data mules are introduced that have large storage capacities and renewable power. However, they act only as relays and take the collected data to the stationary sinks. In [21], the optimal movement of the sink and its sojourn time at different points in a non-clustered network are studied to maximize the network lifetime. In [4], SEAD is proposed in which mobile sinks communicate with sensors using a dissemination tree. Although it is shown that SEAD outperforms directed diffusion [22] and TTDD [18] in terms of energy conservation, it does not consider the effect of MAC nor clustering on the system performance.

The second category of the research on mobile sinks (usually termed as SENMA) assumes that mobile sinks have a complete or nearly complete coverage over the network, and direct communication between sensors and the sink is possible [11][8][23]. Since the network model in this paper assumes direct sink access, it is similar to the network model in the second category. However, our network model does not necessarily require the mobility of the data sink. Within the second category, various issues have been investigated including QoS information retrieval [13], source reconstruction [24], multiple mobile access points [10], and the capacity of cooperative sensor networks [25][10]. While these problems may seem rather different, all of them are mainly related to MAC. Although many MAC protocols have been proposed for conventional flat and clustered ad hoc sensor networks [26][27][2], they may not be suitable for our direct sink access scenario. While in traditional communication networks packet arrival forms a continuous random process, it is not the case in our network setup. In fact, if a sensor is functioning properly, it may have only one packet to transmit when the sink visits the network [13]. This fact along with the energy efficiency motivates designing new MAC protocols.

In [8], nodes use opportunistic ALOHA (O-ALOHA) to access the channel in a non-clustered architecture. In our work, we assume a much simpler MAC to show how clustering enhances the network performance. In [13], a different scheme is proposed where the mobile sink in each time slot notifies a group of nodes to transmit in the next timeslot. Although this scheme achieves high throughput, it requires that most nodes listen to the channel most of the time. In addition, [13] does not consider the effect of clustering nor data aggregation. In our scheme, each node contends for the channel independently

of the other nodes and without direct scheduling by the data sink. In [23], the authors propose a retransmission algorithm that uses a Synchronized, Shared Contention Window (SSCW). SSCW is used to schedule transmissions within a cluster. Their work, in contrast to ours, does not focus on data correlation or the sensors clustering.

On the other hand, many clustering protocols have been proposed for wireless sensor networks [3][15][28] [5]. Specifically in [15], hierarchical clustering is proposed, and the optimal clustering parameters are obtained to minimize the total energy consumption. In more recent works, the authors of [16] and [14] have studied the effect of correlation on clustering. However, they assume that the underlying MAC is contention and error free. Hybrid Energy Efficient Distributed clustering algorithm (HEED) has been proposed in [5]. HEED is a hybrid algorithm, i.e., CHs are randomly selected based on their residual energy level, and nodes join clusters such that a cost function is minimized. The cost function is defined based on the number of power levels that a node can use to reach its CH. Topology Adaptive Spatial Clustering (TASC) is proposed in [29]. TASC allows uniform sampling within clusters, and therefore, better utilization of data correlation. TASC, however, focuses on extracting topological regularities. Clustered AGgregation (CAG) is proposed in [30]. In CAG only one sample per cluster is transmitted up to the aggregation tree. The study of CAG is based on a simple correlation model and single-hop analysis, and does not focus on the clustering aspect of our problem. Theoretical aspects of correlated data gathering has been studied in [31]. It is shown that the optimal clustering becomes NP-complete even in simple cases. Similarly, the authors of [32] have studied lossy gathering of correlated data.

In terms of architecture, the work in [9] is perhaps the nearest research to ours in this paper. More specifically, the authors of [9] consider a heterogeneous sensor network with an aircraft acting as the data sink, and CHs having different hardware-energy characteristics than the ordinary sensors. They provide analytical results to determine the optimal densities for ordinary sensors and CHs minimizing the overall cost of network implementation while assuring a certain network lifetime and network connectivity. However, they assume an ideal MAC, and assume that all data in a cluster can be compressed into one packet, and therefore, they consider a very simple correlation model.

In this work, using the same algorithm as in [15] to generate clusters, we consider a cross-layer approach whereby we are able to consider the effect of MAC on clustering performance in the direct sink access scenario. In addition, we consider a general correlation model [14] to study how correlation affects the optimal cluster sizes and the system performance in terms of latency and energy consumption. We base our design on simplicity to ensure that our protocol requires very limited information about the network status.

### III. REACHBACK CHANNEL CHARACTERISTICS AND AGGREGATION MODEL

Before describing the details of our network architecture and protocol, we first explain our assumptions for the reachback

channel, and describe our data aggregation model.

#### A. Reachback Channel Model

We assume that the data sink has multi-packet reception (MPR) capability and assume that time is slotted. In [12], a general model for channels with MPR capability is developed, which we assume in this paper. In particular, we assume that the following stochastic matrix  $\mathbf{R}$  summarizes the MPR property:

$$\mathbf{R} = \begin{pmatrix} r_{10} & r_{11} & 0 & 0 & \dots \\ r_{20} & r_{21} & r_{22} & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad (1)$$

where  $r_{nk}$  is the probability of  $k$  successful receptions when  $n$  packets are transmitted in one timeslot.

To determine the entries of  $\mathbf{R}$ , we need to consider the physical layer specifications. We emphasize that our protocol and analysis do not depend on the details of how  $\mathbf{R}$  is obtained. In general,  $\mathbf{R}$  can be considered as an abstraction of many system parameters such as transmitter-receiver distance distribution, modulation scheme, and channel fading characteristics. For the purpose of illustration, we use the same method taken in [33] to obtain  $r_{nk}$ 's and assume that CDMA is used to access the channel.

Let  $S$  and  $\sigma^2$  denote the spreading gain and the power of additive white Gaussian noise, respectively. Assuming that each packet consists of  $N_b$  bits and up to  $t$  errors in a packet can be corrected by a block error control code,  $r_{nk}$  is given by [33]

$$r_{nk} = B(k, n, \sum_{j=0}^t B(j, N_b, Q(\sqrt{\frac{3S}{n-1+3S\sigma^2}}))), \quad (2)$$

where  $B(\cdot, \cdot, \cdot)$  denotes the binomial pmf and  $Q(\cdot)$  is the complementary error function [34].

Although (2) assigns a positive probability for  $r_{nk}$ 's for all  $k \leq n$ , in practice only a limited number of packets can be processed and received correctly in a given timeslot and thus,  $k$  is uniformly upper bounded for all values of  $n$  due to hardware capacity constraint.

#### B. Aggregation Model

Let  $\mathbf{X} = (x_1, \dots, x_n)$  be the vector of  $n$  samples of the measured random field returned by  $n$  sensors within a cluster. Let  $\hat{\mathbf{X}}$  be a representation of  $\mathbf{X}$ , and  $d(\hat{\mathbf{X}}, \mathbf{X})$  be a distortion measure. It has been shown that the minimum number of bits required to represent  $\mathbf{X}$  subject to a bound on the total distortion, i.e., when  $E(d(\hat{\mathbf{X}}, \mathbf{X})) \leq D$ , can be computed by the following formula [35]

$$R(D) = \min_{f(\hat{\mathbf{X}}|\mathbf{X}):E(d(\hat{\mathbf{X}}, \mathbf{X})) \leq D} I(\mathbf{X}, \hat{\mathbf{X}}), \quad (3)$$

where  $I(\mathbf{X}, \hat{\mathbf{X}})$  is the mutual information between  $\mathbf{X}$  and  $\hat{\mathbf{X}}$ . When the distortion measure is Mean Square Error (MSE), i.e., when  $d(\mathbf{X}, \hat{\mathbf{X}}) = \|\mathbf{X} - \hat{\mathbf{X}}\|^2$ , a Gaussian source is the worst case and needs the most number of bits to be represented compared with other types of sources [36]. Therefore, for the

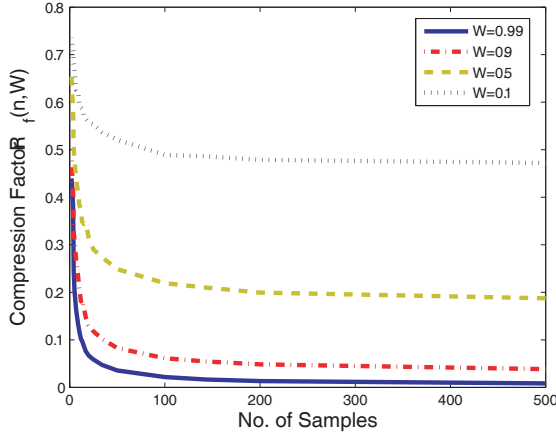


Fig. 2. Compression factor for  $\lambda_0 = 6.25 \text{ m}^{-2}$ .

purpose of illustration, we consider the case where  $\mathbf{X}$  is a multivariate Gaussian random vector, i.e.,  $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ , and assume  $\Sigma_{ii} = \sigma^2 = 1$ . Here  $\Sigma$  is the correlation matrix, and as an example, we consider the case where the correlation falls as a Gaussian function that can be converted to  $f(p_i, p_j) = W \|p_i - p_j\|^2$ , where  $W < 1$ ,  $p_i$  and  $p_j$  are two points in the 2D plane, and  $\|p_i - p_j\|$  is their Euclidean distance. Thus,  $W$  is a quantitative measure for the amount of correlation between spatial samples. We also assume that the total distortion is given by  $D = nD_0$ , where  $D_0$  is the average distortion per sample.

It can be shown that, given a constraint on the total amount of distortion

$$E(\|\mathbf{X} - \hat{\mathbf{X}}\|^2) \leq D = nD_0,$$

the minimum number of bits required to represent a Gaussian source,  $\mathbf{X}$  in our case, is [37]

$$R(D) = \sum_{n=1}^N \frac{1}{2} \log \frac{\lambda_n}{D_n}, \quad (4)$$

where  $\lambda_1 \geq \lambda_2 \dots \geq \lambda_N$  are the ordered eigenvalues of the correlation matrix  $\Sigma$ , and  $D_n$ 's are chosen such that

$$\sum_{n=1}^N D_n = D, \quad D_n = \begin{cases} \theta, & \text{if } \theta < \lambda_n \\ \lambda_n, & \text{otherwise.} \end{cases}$$

Our analysis in the following sections requires the amount of compression as a function of the number of available samples in a cluster. When data is not compressed, each sensor needs  $0.5 \log D_0^{-1}$  bits to encode its data for an MSE of  $D_0$  [37]. Therefore, the compression factor is given by  $\frac{R(D)}{n \cdot 0.5 \log D_0^{-1}}$ . Although it is possible to derive some asymptotic results for  $R(D)$  in a random topology [36], these asymptotic results are not sufficient for our purposes. Thus, we have used the Monte Carlo method whose input is the random node locations subject to node density  $\lambda$ . For a given number of nodes  $n$ , we randomly distribute nodes in an area  $A = (\frac{n}{\lambda})$ . We derive the number of bits according to (4) and find the mean value of  $R(D)$  over different realizations.

Let  $R_f(n, W)$  be the average compression factor obtained

by dividing the average value of  $R(D)$ , using the above method, by  $n \cdot 0.5 \log D_0^{-1}$ . In Fig. 2, we have shown  $R_f(n, W)$  for  $D_0 = 0.01$  and different values of  $W$ . Note that the curves are obtained for a specific node density  $\lambda_0 = 6.25 \text{ m}^{-2}$ . A simple calculation shows that a correlation factor of  $W'$  for a new node density  $\lambda_1$  will lead to the same compression performance if  $W' = W \frac{\lambda_1}{\lambda_0}$ . Note that the special case of  $W = 1$  represents the case where an ideal aggregation is possible, i.e., an arbitrary number of packets can be compressed down into one packet [3][15][14]. Examples of such aggregation include finding the maximum, minimum, or average of the collected information. In this work, we consider general values of  $W$ .

#### IV. NETWORK ARCHITECTURE AND PROTOCOL DESCRIPTION

We assume that  $N_t$  sensors are randomly deployed over a remote area  $A$ , e.g. a desert or a forest, and sensors periodically sense the environmental parameters, e.g. temperature, and record the sensed data. For the simplicity of discussion, we assume that the area  $A$  is circular with radius  $R$ . As mentioned earlier, the mobility of the data sink is not required for our results to hold. However, in the rest of this paper, to describe our protocol, we assume a mobile data sink. Hence, we assume that the data sink, e.g., an airplane flying at height  $H$ , periodically visits the network and initiates one round of data collection to gather the sensed data.

##### A. Cluster Formation

When the data sink visits the network, it broadcasts a beacon packet which initiates data communication, and if required cluster formation. In this paper, we assume that sensors adjust their power and directly communicate with the CHs in one hop [3][23]. Although multihop clusters are a natural extension for regular sensor networks, multihop clustering may be against the purpose of sensor networks with direct sink access. This is because multihop clusters require sensors to perform many network control operations, and one main reason to introduce direct sensor-sink communication is to shift the expensive task of routing, maintaining routing tables, and network control towards data sinks [8][10]. Our proposal that allows direct communication between sensors and CHs adds minimal complexity to the network architecture. In addition, multihop clustering increases the latency of data collection and complicates the MAC design. One other disadvantage of multihop clustering is that in the physical layer synchronization becomes difficult [10].

In CDC-DSA, cluster formation is triggered by the data sink. To ensure that the load of being a CH is rotated among all nodes, cluster formation is performed every multiple rounds of data collection. To construct clusters, each node selects itself as a CH with a fixed probability  $p_c$ . In the following sections, we show how this probability affects the protocol performance. If a node becomes a CH, it broadcasts an advertisement packet (ADV), using a randomized medium access scheme, to announce its status as well as the unique PN code that should be used by all of its cluster-members for data communication. Based on the received signal strength of ADV packets, each

node approximates its distance to the nearby CHs and joins the cluster of the nearest CH.

### B. Intra-Cluster Communication

After reception of the beacon from the data sink, communication starts in the clusters. We assume that, as a part of cluster formation, CHs become aware of their own cluster membership. Each CH then uses this information to setup and broadcast a TDMA schedule for its cluster. To avoid inter-cluster interference, sensors within a cluster use the same PN code broadcasted by their CH. To become synchronized, nodes use the beacon broadcasted by the data sink.

Using TDMA for intra-cluster communication is mainly motivated by considering the following facts. First, since every sensor has exactly one fixed-length packet to send in each round, using a deterministic schedule does not waste bandwidth and at the same time prevents collisions, and therefore, can be more efficient than a random access scheme. Second, note that according to our protocol, cluster formation is performed every multiple rounds of data collection. Therefore, the same TDMA schedule is maintained for several rounds of data collection, and the cost and delay of cluster formation and TDMA schedule set-up will be negligible compared to those for actual data communication. Finally, note that even if a random access scheme, e.g. modified ALOHA, is used, the intra-cluster delay most likely will be proportional to the size of a cluster, and hence our subsequent analysis and results based on TDMA still provide proper design guidelines.

When all data is received by a CH, it performs data aggregation and contends for the reachback channel to forward the data up to the data sink. In CDC-DSA, CHs use CDMA to communicate with the data sink via the MPR physical interface of the data sink. Since the number of nodes in a cluster is a random variable, the aggregated data by a CH may have a variable size (except for the case of  $W = 1$ ). From a practical point of view, since sensors are aimed to be very simple and cheap devices, they may not be able to generate packets with variable sizes or at most they can choose between a few different modes. Moreover, having variable packet sizes complicates the MAC design. Thus, in this paper we assume that the network designer chooses a proper packet size for the aggregated data based on the correlation level between data samples and the average cluster size.

The above discussion suggests that in CDC-DSA the size of a data packet is determined by the type of communication. If the involved communication is within a cluster, then the corresponding packet has the size of a regular data packet, whereas if the communication is between a CH and the data sink, then the packet has the predetermined size needed to accommodate the aggregated data (later given by (8)). Likewise, the length of a timeslot is determined by its nature of communication.

### C. Reachback Medium Access

We assume random access for the reachback channel. Random access scales better than TDMA with the network size, and does not require a centralized scheduler [8][24][23].

TDMA and centralized scheduling would need the knowledge of all CHs' IDs, which can greatly increase the protocol overhead. In addition, the optimal usage of the MPR capability of the sink necessitates concurrent transmissions that could be received with error with a non-negligible probability. Thus, using a fixed transmission schedule as in TDMA would not work efficiently here. Moreover, the performance of TDMA or direct scheduling is more vulnerable to node failures that might happen due to sensor energy shortage. Finally, time synchronization, which is hard to maintain in sensor networks spread over a large area, can significantly affect TDMA performance.

Our random MAC scheme is slightly different from other conventional models, e.g., ALOHA. While in conventional models, traffic arrival is a continuous random process, in our scenario every CH has one packet to transmit, and after this packet is received correctly, the CH does not have any other packets to send until the next round of data collection. This assumption implies that the arrival process can not be modeled by a stochastic Poisson process anymore. Therefore, rather than focusing on steady state analysis, we have to focus on transient analysis.

In CDC-DSA, after a CH finishes data collection, it transmits the aggregated data with probability  $p_n$  to the data sink, where  $n$  represents the number of contending CHs in the current timeslot. If this packet is not received correctly, then the CH retransmits the packet with a new probability in the next timeslot depending on the updated number of contending CHs. To utilize the MPR channel capability, we choose the optimal probability of transmission,  $p_n$ , to be such that the data collection latency is minimized. Details of how to choose  $p_n$  is provided in the next section.

We mentioned earlier that the cluster formation is triggered by the data sink. Therefore, in the ideal scenario, the data sink has the status of all CHs, e.g. by receiving ADV packets broadcasted by each CH. This mechanism gives the necessary information about the number of CHs. Then, based on the correctly demodulated data, the data sink can update the number of remaining CHs. This updated number should be fed back to the CHs and is used as  $n$  by CHs to find the corresponding  $p_n$ . As well as the updated information, the feedback also contains the acknowledgements of correctly received packets.

In practice, the exact number of CHs may not be available. However, as shown in Subsection V-C, the protocol still works if only a fixed rough estimation, e.g. the expected value, of total number of CHs is known a priori. The same is true if a randomly corrupted version of the number is known to the data sink. Both cases reflect the most practical situations, and substantially reduce the MAC complexity. Specifically, the later models the case where non-accurate information is provided by the clustering process.

## V. PROTOCOL ANALYSIS

In this section, we provide an analytical framework to gain insights into the performance of CDC-DSA. We use standard Markov chain analysis to study the packet reception by the

data sink. We show that the optimal  $p_n$ 's can be obtained via dynamic programming and further propose an alternate solution for  $p_n$ 's leading to almost the same latency obtained by the optimal  $p_n$ 's. In addition, we discuss how inaccurate information increases the average latency, and finally, using random geometry tools, we obtain an expression for energy consumption and an approximate upper bound for the data collection latency in the presence of arbitrary correlation levels.

### A. Markov Chain Analysis

If clusters are of equal size, then they ideally finish their TDMA schedule at the same time and try to contend for the channel at the same next timeslot. This means that the number of remaining CHs, whose packets are not received by the data sink, is equal to the number of contending CHs. In a practical situation, because clusters are formed randomly, they have different sizes. Consequently, their corresponding CHs finish their TDMA schedule and start to contend for the channel at different timeslots. As a result, the number of remaining CHs, which is the available information based on which CHs choose  $p_n$ , is not necessarily equal to the number of contending CHs in a given time slot. However, as the data collection from CHs continues, the number of contending CHs approaches to the number of remaining CHs. We base our analysis on equal size clusters. Later in Section VI, we show that our analysis predicts the total energy consumption very well when the clusters are formed randomly according to our protocol.

Based on the above assumption, it can be seen that the number of contending CHs forms a finite discrete Markov chain. Consider the transition from the state ( $m$ ) to the state ( $n$ ) and let  $t_{mn}$  denote the probability of that transition. Conditioned on  $i$  CHs transmitting in the next timeslot,  $m - n$  packets should be received correctly by the data sink to produce such a transition. Therefore, considering all possibilities, we have

$$t_{mn} = \begin{cases} \sum_{i=m-n}^m B(i, m, p_m) r_{i, m-n} & n \leq m, m \neq 0 \\ 0 & n > m \\ 1 & n, m = 0 \end{cases}.$$

Clearly, there is one absorbing state, (0), in the Markov chain. Given that the current state is ( $m$ ), we can calculate the mean time to absorption,  $a_m$ , and the mean number of transmissions,  $u_m$ , until absorption. Let

$$\mathbf{a} \triangleq [a_1, \dots, a_{N_t}], \quad \mathbf{u} \triangleq [u_1, \dots, u_{N_t}].$$

Defining  $\mathbf{W}$  as the transition matrix of all transient states, we have [38]

$$\mathbf{a} = (\mathbf{I} - \mathbf{W})^{-1} \mathbf{1}, \quad (5)$$

where  $\mathbf{I}$  is the identity matrix and  $\mathbf{1}$  represents a vector with all entries equal to 1. To compute  $\mathbf{u}$ , we assume that the current state of the chain is  $N$ . In this case,  $n$  packets are transmitted with probability  $B(n, N, p_n)$ , and the current state will transit to the state  $N - i$  with probability  $r_{ni}$ . Considering all possible

values for  $n$ , we have

$$\begin{aligned} u_N &= \sum_{n=0}^N B(n, N, p_N) n + \sum_{n=0}^N \sum_{i=0, i \neq N}^n B(n, N, p_N) r_{ni} u_{N-i} \\ &= N p_N + \sum_{K=1}^N t_{NK} u_K. \end{aligned}$$

Summarizing the above equation for all values of  $N$ , i.e. all states, in a vector format, we get

$$\mathbf{u} = \mathbf{v} + \mathbf{W}\mathbf{u}, \quad \mathbf{u} = (\mathbf{I} - \mathbf{W})^{-1} \mathbf{v}, \quad (6)$$

where,  $\mathbf{v} = [1p_1, \dots, ip_i, \dots, N_t p_{N_t}]$  is the vector of the average number of transmissions for any current state.

The values given by (5) and (6) are conditioned upon the current state. To find the unconditional expected number of transmissions  $E(Tr)$  and timeslots  $E(Ts)$  until absorption, we need to choose a proper size for the packets accommodating the aggregated data (as discussed in Section IV-B) and find the initial distribution of the Markov chain. Let  $P_l$  be the length of the aggregated data in a cluster as a function of  $W$  and  $n_c$ , where  $n_c$  denotes the number of nodes in the cluster. In this analysis, we use  $E(P_l(W, p_c))$  as the proper packet size. Since the distribution of the number of nodes in a cluster is not known, we cannot find a useful closed-form solution for  $E(P_l(W, p_c))$ . However, by considering [39], we have

$$E(n_c) = p_c^{-1}, \quad (7)$$

and can approximate  $E(P_l(W, p_c))$  by

$$E(P_l(W, p_c)) \simeq E(n_c) R_f(E(n_c), W) = \frac{1}{p_c} R_f\left(\frac{1}{p_c}, W\right), \quad (8)$$

where  $R_f$  is the compression factor, and has been defined in Section III-B.

Assuming that each node selects itself as a CH independently of the other nodes, the actual number of CHs in the network has a binomial distribution and so does the initial condition of the Markov chain:

$$P(N_0 = n) = B(n, N_t, p_c). \quad (9)$$

Now, we can calculate  $E(Tr)$  and  $E(Ts)$  by

$$\begin{aligned} E[Tr] &= B(0, N_t, p_c) \frac{\mathbf{u}(N_t)}{E[P_l(W, p_c)]} \\ &+ \sum_{n=1}^{N_t} B(n, N_t, p_c) \mathbf{u}(n), \end{aligned} \quad (10)$$

$$\begin{aligned} E[Ts] &= B(0, N_t, p_c) \frac{\mathbf{a}(N_t)}{E[P_l(W, p_c)]} \\ &+ \sum_{n=1}^{N_t} B(n, N_t, p_c) \mathbf{a}(n). \end{aligned} \quad (11)$$

The first terms in the above formulas come from the fact that if there is no CH in the network, then all nodes send their data directly to the data sink. In such a case, nodes use the original packet sizes. Therefore, in (10) and (11) we divide the first terms by  $E[P_l(W, p_c)]$  in order to make  $E(Tr)$  and  $E(Ts)$  correspond to the case of using a timeslot required to

transmit a packet with size  $E[P_i(W, p_c)]$ .

### B. Selection of Transmission Probabilities

Our goal in this subsection is to find the optimal transmission probabilities to minimize the delay of data collection from CHs. To start, note that the absorbing times in (5) can be written as

$$a_n = 1 + \sum_{i=1}^n t_{ni} a_i. \quad (12)$$

Using a backward recursion dynamic programming approach [40], we can find the optimal  $p_n$  for each value of  $n$  by

$$p_n^* = \arg \min_{0 \leq p \leq 1} \frac{1}{1 - t_{nn}} \left( 1 + \sum_{i=1}^{n-1} t_{ni} a_i \right). \quad (13)$$

Let  $a_n^*$  be the corresponding optimal value for  $a_n$ . Although dynamic programming can be used to find  $p_n^*$  for different values of  $n$ , it does not provide useful insights into how the MAC should be designed when  $n$  becomes large. In the following, we approach the problem from a different angle and derive an alternate solution which leads to almost the same set of  $a_n^*$ 's obtained by the solution of dynamic programming.

Let  $\lambda_n^* = np_n^*$ , and each possible set of  $\{p_n^*\}_{n=1}^\infty$  define a policy. To gain more insight on the behavior of  $p_n^*$  as  $n \rightarrow \infty$ , we define a stationary policy to be a policy  $\xi$  with  $\{p_n^\xi\}_{n=1}^\infty$  for which  $\lim_{n \rightarrow \infty} np_n^\xi = \lim_{n \rightarrow \infty} \lambda_n^\xi = \lambda_\infty^\xi < \infty$ . The existence of  $\lambda_\infty^\xi$  for the policy  $\xi$  ensures that the binomial distribution of the number of transmissions approaches to a Poisson distribution with rate  $\lambda_\infty^\xi$  (as  $n \rightarrow \infty$ ). Let  $t_i^n = t_{n, n-i}$ . Since  $t_{mn}$ 's are non-negative and bounded above, we have

$$t_i^\infty \triangleq \lim_{n \rightarrow \infty} t_i^n = \lim_{n \rightarrow \infty} t_{n, n-i} = \sum_{j=i}^{\infty} e^{-\lambda_\infty^\xi} \frac{(\lambda_\infty^\xi)^j}{j!} r_{ji}. \quad (14)$$

As mentioned earlier in Subsection III-A, in practice the maximum number of packets received in one channel access trial,  $n_0$ , is finite, and is determined by hardware capacity constraint. Thus, we can assume that  $r_{ji} = t_i^n = 0$ , for  $i > n_0$ . Let  $a_n^\xi$ ,  $n \geq 1$  be the solution to (12) using  $\xi$ , and  $\rho_\xi$  be defined as

$$\rho_\xi = \left( \sum_{i=0}^{n_0} i t_i^\infty \right)^{-1} = \left( \sum_{i=1}^{\infty} e^{-\lambda_\infty^\xi} \frac{(\lambda_\infty^\xi)^i}{i!} C_i \right)^{-1},$$

where  $C_i = \sum_{k=1}^{\min(i, n_0)} k r_{ik}$ .

Finally, let  $\Xi$  be the set containing all stationary policies, and  $\rho^*$  and  $\lambda_\infty^*$  be defined as

$$\lambda_\infty^* = \arg \max_{\lambda > 0} \sum_{i=1}^{\infty} e^{-\lambda} \frac{\lambda^i}{i!} C_i,$$

$$\rho^* = \left( \max_{\lambda > 0} \sum_{i=1}^{\infty} e^{-\lambda} \frac{\lambda^i}{i!} C_i \right)^{-1}.$$

We define a stationary policy  $\xi^*$  to be optimal if there is no other policy in  $\Xi$  which asymptotically outperforms  $\xi^*$ . We have the following proposition.

*Proposition 1:* Let  $\xi$  be a stationary policy. We have

- $\lim_{n \rightarrow \infty} \frac{a_n^\xi}{n \rho_\xi} = 1$ .
- The optimal stationary policy  $\xi^*$  is the one for which we have  $\lambda_\infty^{\xi^*} = \lambda_\infty^*$  and  $\rho_{\xi^*} = \rho^*$ .

*Proof:* Consider a stationary policy  $\xi$  with  $\lambda_\infty^\xi$ . In the rest of the proof, to simplify the notation, we drop the dependency of variables on  $\xi$ . Considering (14) and the definition of  $n_0$ , and letting  $a_n^s$  to be a shifted version of  $a_n$  by an amount of  $\tau + 1$ , i.e.,  $a_n^s = a_{n+\tau+1}$ , in the limit of large  $\tau$ , we can rewrite (12) as

$$\hat{a}_n^s = 1 + \sum_{i=0}^{n_0} t_i^\infty \hat{a}_{n-i}^s. \quad (15)$$

We use the notation  $\hat{a}_n^s$  to indicate that the following analysis holds in the limit of large  $\tau$ , i.e., when, for  $n \geq \tau + 1$ , we have  $t_i^n = t_i^\infty$ ,  $0 \leq i \leq n_0$ .

The equation in (15) is a linear difference equation which can be solved uniquely given the initial conditions  $\{\hat{a}_i^s\}_{i=-n_0}^{-1}$ , and considering the input to be  $x(n) = u(n)$ , where  $u(n)$  is the step function. Taking the z-transform of both sides of (15) leads to

$$\hat{A}(z) = \frac{1}{1 - z^{-1}} + \sum_{i=0}^{n_0} t_i^\infty (z^{-i} \hat{A}(z) + \sum_{k=1}^i \hat{a}_{k-1-i}^s z^{-k+1}).$$

Let  $B(z) = 1 - \sum_{i=0}^{n_0} t_i^\infty z^{-i}$ . The above equation can be simplified to

$$\hat{A}(z) = \frac{1}{(1 - z^{-1})B(z)} + \frac{(\sum_{i=0}^{n_0} t_i^\infty \sum_{k=1}^i \hat{a}_{k-1-i}^s z^{-k+1})}{B(z)}.$$

$B(z)$  can be rewritten as

$$\begin{aligned} B(z) &= \sum_{i=0}^{n_0} t_i^\infty (1 - z^{-i}) \\ &= (1 - z^{-1}) \sum_{i=0}^{n_0} t_i^\infty \left( \sum_{k=0}^{i-1} z^{-k} \right). \end{aligned}$$

Therefore, we can conclude that  $B(z)$  has a first order root at  $z = 1$ . On the other hand, assuming  $n_0 \geq 1$ , for all values of  $z \neq 1$  with  $|z| \geq 1$ , we have  $|\sum_{i=0}^{n_0} t_i^\infty z^{-i}| < \sum_{i=0}^{n_0} t_i^\infty |z^{-i}| \leq \sum_{i=0}^{n_0} t_i^\infty = 1$ . Thus  $B(z)$  cannot have any other root on or outside the unit circle. This ensures that in  $\hat{a}_n^s$ , all terms associated with the other poles except for  $z = 1$  will eventually die out as  $n \rightarrow \infty$ . Finding the residue for the second order pole at  $z = 1$ , we have

$$\hat{A}(z) = \frac{1}{(1 - z^{-1})^2 (\sum_{i=0}^{n_0} i t_i^\infty)} + \Gamma(z),$$

where  $\Gamma(z)$  is the remaining part of  $\hat{A}(z)$  containing all the other poles of  $\hat{A}(z)$  as well as the first order contribution of the pole at  $z = 1$ . Taking inverse z-transform, we have

$$\hat{a}_n^s = \frac{n+1}{\sum_{i=0}^{n_0} i t_i^\infty} u(n+1) + \gamma(n), \quad (16)$$

where  $|\lim_{n \rightarrow \infty} \gamma(n)| = \alpha < \infty$ . Since  $\hat{a}_n^s$  is just a shifted version of  $a_n$  in the limit of large  $\tau$ , (16) implies that when  $t_i^n = t_i^\infty$ ,  $0 \leq i \leq n_0$ ,  $a_n$  is asymptotically a linear function with slope  $\rho_\xi$ . In what follows, we show that assuming  $t_i^n \rightarrow$

$t_i^\infty$  as  $n \rightarrow \infty$ ,  $0 \leq i \leq n_0$ , is sufficient to prove that  $a_n$  is indeed an asymptotically-linear function with slope  $\rho_\xi$ . First, note that (14) implies that for any given  $\epsilon > 0$ , we can find  $n_\epsilon$  such that for  $n > n_\epsilon$ , we have  $\sum_{i=0}^{n_0} |t_i^n - t_i^\infty| < 2\epsilon$ . This allows us to consider two extreme density functions, which provide lower and upper bounds for  $a_n$ . More specifically, let  $t_i^l$ ,  $0 \leq i \leq n_0$ , be given by

$$t_0^l = t_0^\infty - 2\epsilon, \quad t_i^l = t_i^\infty, \quad 0 < i < n_0 - 1, \quad t_{n_0}^l = t_{n_0}^\infty + 2\epsilon,$$

and  $t_i^u$ ,  $0 \leq i \leq n_0$ , be given by

$$t_0^u = t_0^\infty + \epsilon, \quad t_i^u = t_i^\infty, \quad 0 < i < n_0 - 1, \quad t_{n_0}^u = t_{n_0}^\infty - \epsilon.$$

Let  $a_i^u = a_i^l = a_i$  for  $n_\epsilon + 1 - n_0 \leq i \leq n_\epsilon$ . Furthermore, let  $a_i^l$  and  $a_i^u$ ,  $i > n_\epsilon$ , be the sequences obtained by (12) assuming the densities given by  $t_i^l$  and  $t_i^u$ ,  $0 \leq i \leq n_0$ , respectively. Using these definitions, it is not hard to see that

$$a_n^l \leq a_n \leq a_n^u, \quad n > n_\epsilon.$$

Dividing the above by  $n\rho_\xi$ , and applying the limit analysis for the two new densities, we obtain

$$\frac{\sum_{i=1}^{n_0} it_i^\infty}{\sum_{i=1}^{n_0} it_i^\infty + 2\epsilon n_0} \leq \lim_{n \rightarrow \infty} \frac{a_n}{n\rho_\xi} \leq \frac{\sum_{i=1}^{n_0} it_i^\infty}{\sum_{i=1}^{n_0} it_i^\infty - \epsilon n_0}.$$

Since  $\epsilon$  can be chosen arbitrarily small, it follows that

$$\lim_{n \rightarrow \infty} \frac{a_n}{n\rho_\xi} = 1.$$

Thus,  $a_n$  for a stationary policy is asymptotically a linear function, proving part (a) of the proposition. Part (b) of the proposition directly follows from part (a) and the fact that we are interested in asymptotic optimality. ■

Proposition 1 motivates us to select  $p_n$  as

$$p_n = \arg \max_{0 \leq p \leq 1} \sum_{i=0}^n i t_{n,n-i} = \arg \max_{0 \leq p \leq 1} \sum_{i=1}^n B(i, n, p) C_i, \quad (17)$$

which for large  $n$  becomes

$$p_n = \frac{1}{n} \arg \max_{\lambda > 0} \sum_{i=1}^{\infty} e^{-\lambda} \frac{\lambda^i}{i!} C_i = \frac{\lambda_\infty^*}{n}. \quad (18)$$

Note that if there are multiple solutions to (17), we choose the smallest solution in order to minimize the average number of transmissions in each timeslot. If the optimal policy is such that  $\lim_{n \rightarrow \infty} \lambda_n^*$  exists, Proposition 1 determines the limit. In general, however, the limit may not exist. In that case, the following proposition states that the optimal policy asymptotically performs the same as the optimal stationary policy, and thus, stationary policies are sufficient for asymptotic optimality.

*Proposition 2:* Let  $a_n^*$  be the optimal solution to (12). Consider the optimal stationary policy  $\xi^*$ , and let  $\{a_i^{\xi^*}\}_{i=1}^{\infty}$  be the solution to (12) using  $\xi^*$ . We have

$$\lim_{n \rightarrow \infty} \frac{a_n^*}{a_n^{\xi^*}} = 1. \quad (19)$$

*Proof:* See the Appendix. ■

In the sequel, we compare the performance of the two studied policies, the first one obtained by the dynamic prog-

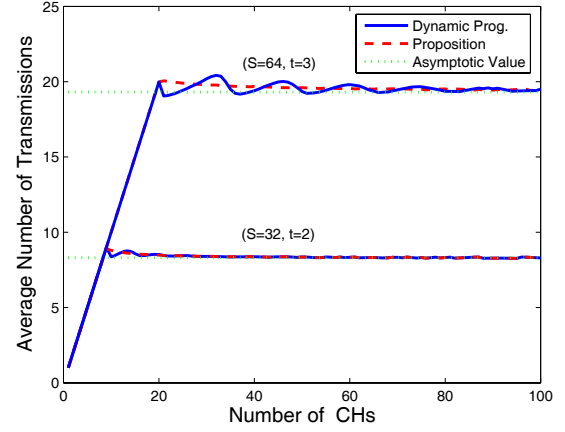


Fig. 3. Average number of transmissions for different spreading gains,  $S$ , and error correcting capabilities,  $t$ .

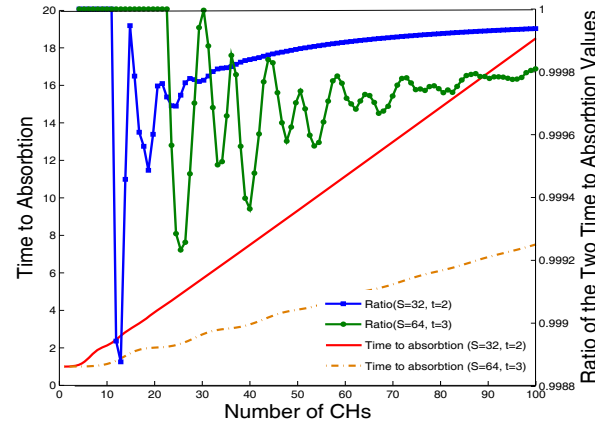


Fig. 4. Expected time to absorption (the latency to collect CHs' data packets) and the ratio of the latency obtained by dynamic programming to the latency obtained by the proposition result.

ramming approach, and the other from the result of Proposition 1, which is specified by (17). In Fig. 3, we have shown the average number of transmissions in each timeslot as a function of the number of contending CHs for the two policies. As expected, the average number of transmissions converges to the predicted asymptotic value,  $\lambda_\infty^*$ , as  $n$  becomes large. Interestingly,  $\lambda_n^*$  shows an oscillatory behavior compared to  $n p_n$  obtained by (17). In addition, for the system with higher error correcting capability and larger spreading gain, the convergence rate is slower but  $\lambda_n^*$  converges to a larger value.

Fig. 4 demonstrates the corresponding time-to-absorption for the optimal policy. On the same figure, we have shown the ratio of  $a_n$ 's obtained by the solution of (13) to the ones using the solution of (17). From the figure, it is evident that the two approaches perform almost the same. This figure also suggests that when the number of CHs is large, the proposed MAC does not require the exact number of contending CHs, implying that our protocol is fairly robust. We elaborate on this topic in the next subsection.



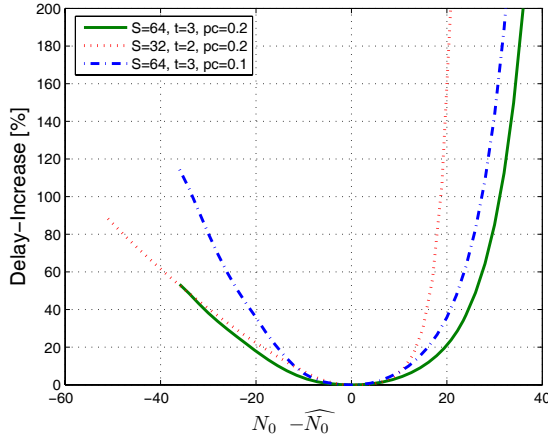


Fig. 5. MPR delay performance without feedback from the clustering process.

The figure also shows that the delay of collecting packets from CHs approaches to a linear function as was predicted by our analysis. One application of this result is that for a given  $p_c$  and therefore, a given cluster size, the overall delay scales linearly with the number of nodes.

### C. Robustness of MAC in CDC-DSA

As discussed in the previous section, to minimize the delay of data collection, the data sink needs to know the total number of CHs in the network,  $N_0$ , in order to update the current number of contending CHs in each timeslot. Here we assume that the data sink can have only partial information about the total number of CHs. We consider two cases:

- In one extreme case, the data sink is only aware of the binomial distribution of  $N_0$  with parameters  $N_t$  and  $p_c$ .
- For the other case, we may reasonably assume that the clustering process will provide some feedback to the data sink. However, since the clustering is a decentralized process, the feedback may be a noisy version of  $N_0$ .

If the first case is assumed, then the data sink can simply use a fixed estimation of  $N_0$  denoted by  $\hat{N}_0$ . Here, we do not assume any adaptive mechanism to improve the estimation. Fig. 5 shows how the delay to collect packets from CHs increases as the actual  $N_0$  deviates from  $\hat{N}_0$  for  $N_t = 1000$ . When  $S = 64$  and  $t = 3$ , we simply choose  $\hat{N}_0 = E(N_0) = N_t p_c$ . For such a simple estimation, the average delay-increase values<sup>2</sup> are only 9.26% and 8.59% for  $p_c = 0.2$  and  $p_c = 0.1$ . Note that when  $N_0$  is more than  $E(N_0)$ , the delay increases faster than the case where  $N_0$  is less than  $E(N_0)$ . This observation suggests that overestimation leads to a better delay performance. This is particularly important for systems with reduced MPR capability. As an example, for  $S = 32$ ,  $t = 2$ , and  $p_c = 0.2$ , using  $\hat{N}_0 = E(N_0) = 200$  leads to more than 60% increase in the average delay, but we can use the above observation to keep the average delay-increase less than 20% by choosing  $\hat{N}_0 = 215$ .

<sup>2</sup>The average delay-increase is simply  $\sum_i p(N_0 = i) D_{inc}(i, \hat{i})$ , where  $D_{inc}(i, \hat{i})$  is the delay-increase when  $N_0 = i$  and  $\hat{i}$  is the estimated version of  $i$ .

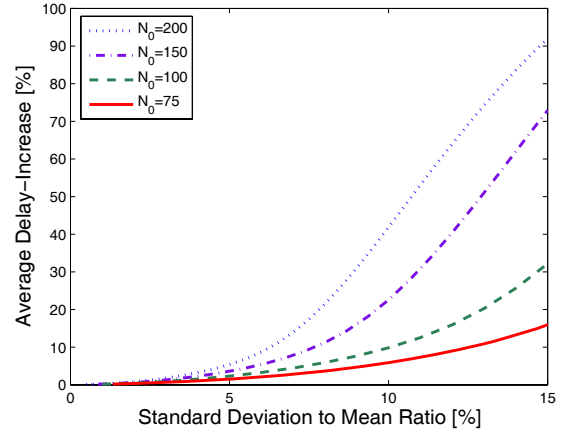
Fig. 6. MPR delay performance with feedback from the clustering process ( $S=64$ ,  $t=3$ ).

Fig. 6 depicts the percentage of the delay-increase for the second case where the clustering process provides the data sink with a feedback. We assume that the feedback is corrupted by a truncated zero-mean Gaussian noise, and hence, the feedback is also a truncated Gaussian random variable with mean equal to  $N_0$ . Fig. 6 depicts the average delay-increase as a function of the standard deviation to mean ratio. It can be seen that a larger  $N_0$  results in a larger average delay-increase. Interestingly, if the noise standard deviation is less than 8% of  $N_0$ , then the delay-increase is less than 20% for all cases. However, in practice, the number of clusters is rarely more than 100 and in such cases, a noise standard deviation as large as 10% of  $N_0$  increases the delay only 10%.

Results of the last two figures indicate that if small increases of the delay are acceptable, it is not necessary for the protocol to have access to the exact number of CHs. This implies that small deviations from  $p_n$  has little impact on the performance and consequently our protocol is fairly robust to small errors in  $p_n$ .

### D. Energy Consumption Analysis

Our goal here is to obtain the expected value of the total energy consumption,  $E[U]$ , needed to gather data within clusters and transmit data to the data sink.

Let  $E_{DP}$  be the energy required to process a bit for data aggregation purposes and  $E(N_0) = N_t p_c$  represent the average number of clusters. Since we are interested in the average energy consumption, in our computations, all fading components average out except the path loss fading that would be the only effective component. Therefore, we assume that if a node wants to transmit a packet to another node at a distance  $d$ , it should consume  $c_0 + c_1 d^\alpha$  units of energy [3][5][20][10]. Based on this energy model, let  $E(S_{c_0+c_1|x_i|^\alpha})$  denote the expected value of the total energy consumed by all cluster-members to communicate with their CH. In  $E(S_{c_0+c_1|x_i|^\alpha})$ ,  $|x_i|$  is the distance between the  $i$ th cluster-member and its CH. The effect of correlation can be accounted for by considering

the chosen packet length given by (8). We thus have

$$E[U] = N_b \left( E[P_l(W, p_c)] E[Tr](c_0 + c_1 H^\beta) + E(N_0) E(S_{c_0+c_1|x_i|^\alpha}) + N_t E_{DP} \right), \quad (20)$$

where  $H$  is the height at which the data sink flies over the network and,  $\alpha$  and  $\beta$  show the roll-off factors for the two types of data communication, i.e., within clusters and from CHs to the data sink, respectively. We assume that  $H$  is large enough so that the distance between nodes and the data sink can be approximated by  $H$ . In (20), the first term in braces shows the energy required to communicate with the data sink. The second term represents within-cluster energy consumption, and the last one shows the processing energy expenditure.

To calculate (20), we need to compute  $E(S_{c_0+c_1|x_i|^\alpha})$ . This expectation can be computed using random geometry theory [39]. Based on our assumptions, nodes are uniformly distributed over the area  $A$ . This point process can be approximated by a Poisson point random process whose density is  $\lambda = \frac{N_t}{A}$ . If every node can be a CH with probability  $p_c$ , then it can be shown that the CH point process is also a Poisson process with density  $\lambda_1 = p_c \lambda$  and the remaining points constitute another Poisson process with density  $\lambda_0 = (1-p_c)\lambda$  [38].

Furthermore, in our cluster formation, we have assumed that every node joins the cluster of the nearest CH. Therefore, the area  $A$  is tessellated to a set of Voronoi cells. Let  $\mathcal{C}_0$  represent a typical Voronoi cell whose nucleus is located at the origin,  $\Pi_0$  denote the Poisson point process associated with the non-CH nodes, and  $x_i$  be a member of  $\Pi_0$ . We define a function  $f(x_i)$  as a property of  $x_i$ , e.g., its distance to the CH, and  $S_f$  as the summation of that property over all cluster-members, i.e.,

$$S_f = \sum_{x_i \in \Pi_0} f(x_i) \mathbf{1}\{x_i \in \mathcal{C}_0\},$$

where  $\mathbf{1}\{\cdot\}$  is the indicator function. We can compute the expected value of  $S_f$ ,  $E(S_f)$ , as follows [39]

$$E(S_f) = \lambda_0 \int f(x) \exp(-\lambda_1 \pi |x|^2) dx. \quad (21)$$

In our case,  $f(x_i)$  is given by our energy consumption model, which is  $c_0 + c_1 |x_i|^\alpha$ . For example, when  $\alpha$  is 4, plugging this form of  $f(x_i)$  into (21) leads to

$$\begin{aligned} E(S_{c_0+c_1|x_i|^\alpha}) &= \lambda_0 \int f(x) \exp(-\lambda_1 \pi |x|^2) dx \\ &\simeq \frac{-2\pi \lambda_0 c_0}{2a_0} \exp^{-a_0 r^2} \Big|_0^R \\ &\quad + 2\pi \lambda_0 c_1 \left( \frac{-r^4}{2a_0} - \frac{r^2}{a_0^2} - \frac{1}{a_0^3} \right) \exp^{-a_0 r^2} \Big|_0^R \end{aligned} \quad (22)$$

where  $a_0 = \lambda_1 \pi$ . The approximation in (22) is due to the fact that the area  $A$  is finite.

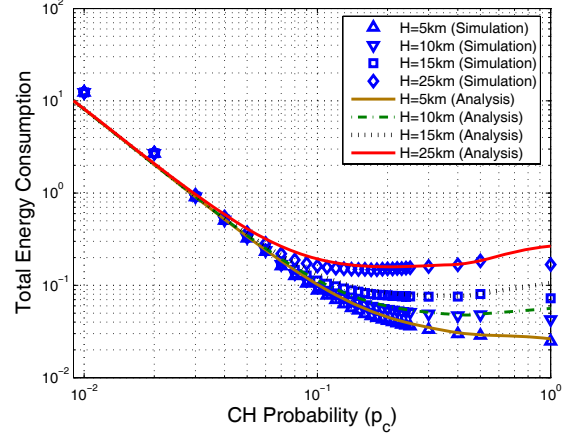


Fig. 7. Energy consumption vs. CH probability ( $\beta = 2$ ,  $R = 1000$  m).

### E. Latency Analysis

In this work, we obtain an approximate upper bound for the total latency of data collection. It is clear that the total latency has two components. The first one is the latency introduced by the TDMA schedules within clusters. The second component is the latency introduced by the contention in the reachback channel. TDMA latency is bounded by the largest cluster size in the network. We assume a maximum cluster size that is three standard deviations above the mean. Recall that  $n_c$  is the number of nodes in a cluster. For the variance of  $n_c$ , we have [39]

$$\sigma_{n_c}^2 = \frac{\lambda_0}{\lambda_1} + 0.28 \frac{\lambda_0^2}{\lambda_1}. \quad (23)$$

On the other hand, for large  $N_0$ , the latency of the reachback channel is maximized when all clusters have equal sizes, i.e., when all CHs contend for the channel at the same time. Therefore,  $E(Ts)$  is an upper bound for the reachback latency. Let  $L^u$  denote our approximate upper bound for latency. Considering the TDMA and the reachback latency bounds, we obtain

$$L^u = E(n_c) + 3\sigma_{n_c} + E[P_l(W, p_c)] E(Ts), \quad (24)$$

where  $E(n_c)$ ,  $E[P_l(W, p_c)]$ ,  $E(Ts)$ , and  $\sigma_{n_c}$  are derived in (7), (8), (11), and (23), respectively.

## VI. SIMULATION RESULTS AND DISCUSSION

In this section we provide our numerical results based on (20) and (24). We justify our analysis by simulating CDC-DSA in a wide range of parameters. Each simulation data point is obtained by averaging over 200 random realizations. Throughout these results,  $N_t = 1000$ ,  $c_0 = 50$  nJ/bit [3],  $c_1 = 0.0013$  pJ/bit/m<sup>4</sup> [3],  $E_{DP} = 5$  nJ/bit/signal [3], and we set  $N_b = 200$  bits. We choose the network radius  $R$ , the spreading gain  $S$ , and the compression factor  $W$  to be 1000 meters, 32, and 0.5 respectively, unless otherwise stated.

Through Fig.7, Fig.8, and Fig.9, we have presented both the analysis and simulation results according to various clustering parameters. These figures suggest that for medium to high

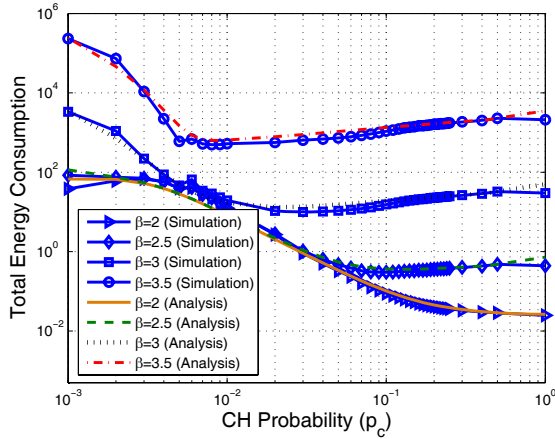


Fig. 8. The effect of roll-off factor on clustering performance ( $H = 5000$  m,  $R = 1000$ m).

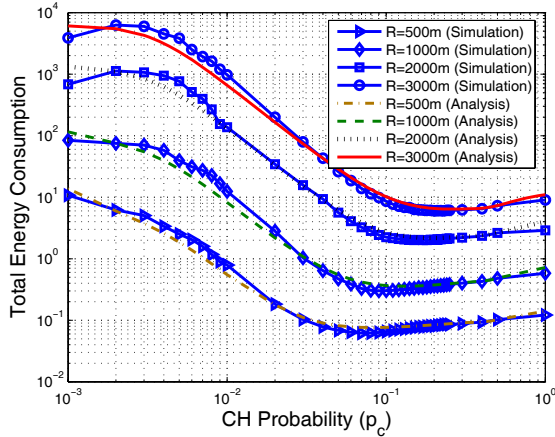


Fig. 9. The effect of network radius on clustering performance ( $\beta = 2.5$ ,  $H = 5R$ ).

values of  $p_c$ , there is a perfect match between our analysis and the simulation. When  $p_c$  decreases, we expect to observe a mismatch due to the finite size of the network. In fact, when  $p_c$  is less than 0.01, there are only about 10 clusters on average. Consequently, the likelihood of observing a typical cluster (with respect to a network with infinite area) decreases. Therefore, our analysis may not be accurate when  $p_c$  is very low. On the other hand, as explained in Section V, unequal cluster sizes can affect our analysis results for energy consumption.

Fig.7 shows the energy consumption as a function of  $p_c$  when  $\beta = 2$ . This figure suggests that when correlation is low, i.e, when  $W = 0.5$ , and  $H$  is relatively small, the gain from clustering vanishes. However, as  $H$  increases, the energy savings from clustering also increases, and the optimal  $p_c$  decreases. This means that larger clusters lead to a better performance in terms of energy consumption. This figure also indicates that at low  $p_c$ , the performance is weakly coupled with  $H$ . In fact, as  $p_c$  decreases, larger clusters are formed and the average distance to the CHs increases. Consequently, the energy consumption within clusters dominates the energy

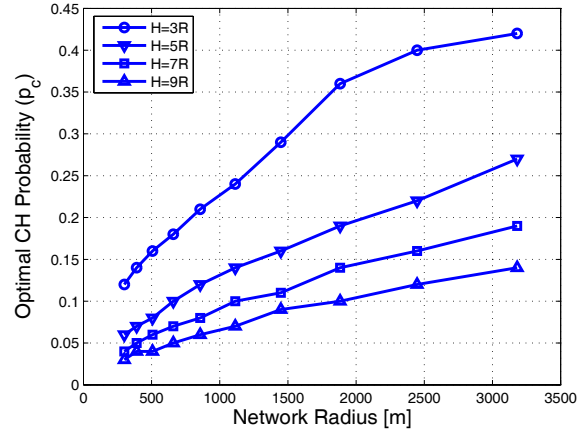


Fig. 10. The effect of network dimensions on optimal CH probabilities ( $\beta = 2.5$ ).

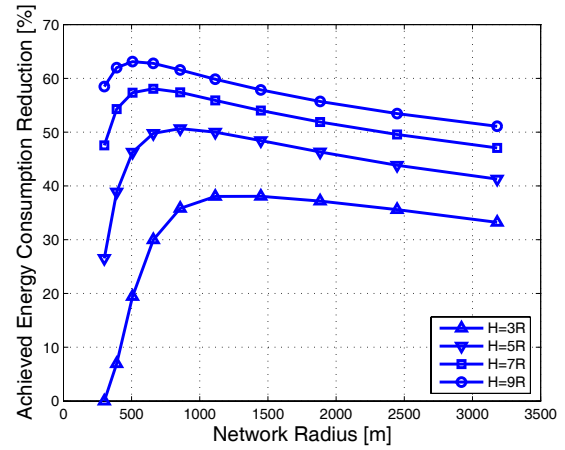


Fig. 11. The amount of energy consumption reduction for different network dimensions ( $\beta = 2.5$ ).

consumed to communicate with the data sink.

In Fig.8, we have demonstrated the effect of roll-off factor on the optimal clustering. As expected, increasing roll-off factor translates to more energy expenditure needed to communicate with the data sink. Therefore, in such cases, larger clusters are more efficient (smaller  $p_c$ ). In addition, it can be seen that the performance is highly sensitive to the changes in the roll-off factor, and the total energy consumption is reduced more than an order of magnitudes when the roll-off factor takes values equal or larger than three.

Fig.9 shows the effect of network radius on clustering performance while the total number of nodes is kept constant at 1000. We set  $\beta = 2.5$  to be more conservative in our results. In addition, we choose  $H = 5R$ . Since a smaller network radius results in smaller distances between nodes, decreasing network radius improves clustering performance and larger clusters are optimal. Therefore, as  $R$  decreases, the optimal  $p_c$  decreases, too. Fig.10 shows the optimal probabilities as a function of network radius with the same parameters as in Fig.9 except that we vary  $H$  from  $3R$  to  $9R$ . It can be observed that while an increase in  $R$  increases the optimal  $p_c$ , increasing

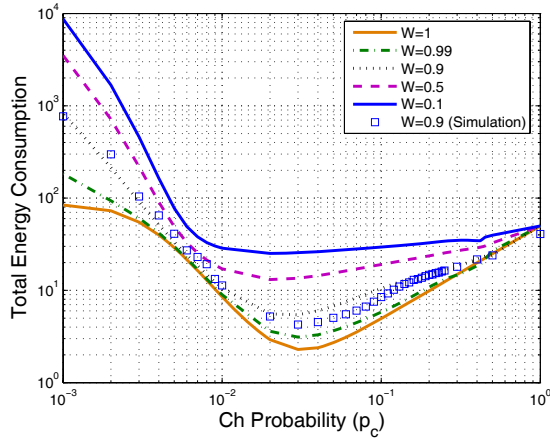


Fig. 12. The effect of correlation on clustering performance ( $\beta=3$ ,  $H = 5000$  m, and  $R = 1000$  m).

$H$  has a reverse effect on the optimal  $p_c$ . The corresponding amount of energy reductions over a data collection scheme with no clustering, i.e.  $p_c = 1$ , for various values of  $R$  and  $H$  is depicted in Fig. 11. It can be inferred that clustering performs much better in moderately large networks.

In Fig. 12, we have shown the energy consumption as a function of the correlation factor  $W$ . For clarity, we have shown the simulation results only for  $W = 0.9$ . As expected, the lower the correlation, the less is the impact of optimal clustering. Furthermore, for low correlation levels the performance is less sensitive to changes in  $p_c$ . This can be explained as follows. When correlation is low, the size of the aggregated data increases almost linearly with the cluster size with a slope comparable to one. This means that even after aggregation, the size of the data needed to be transmitted to the data sink is comparable to the size of the data prior to the aggregation. At the same time, since the average distance from cluster-members to CHs increases, more energy is needed to communicate with CHs. Therefore, at most values of  $p_c$  the gain of data compression is almost canceled by the increase in the energy consumption, and hence, the overall energy consumption becomes less sensitive to the variation in  $p_c$ .

In Fig. 13, we have shown the achievable energy savings for different correlation levels and roll-off factors. Two important observations from this figure are as follows. First, even when correlation is very low (when  $W = 0.1$ ) clustering reduces energy consumption at least 30%. Second, even if the effective roll-off factor is not constant in time due to time variant fading in the physical channel, this figure suggests that the energy reduction is always within a strip of height almost 30%. Consequently, CDC-DSA features substantial energy savings even when time varying fading is present.

Fig. 14 depicts the data collection latency for various values of  $W$  and  $p_c$ . It is clear that  $H$  can change only the propagation delay, which is negligible compared to the queuing delay. Therefore, the results of this figure are approximately independent of  $H$  as well as  $\beta$ . From this figure it is clear that  $L^u$  is a tight bound for most values of  $p_c$ . As expected, a larger correlation results in a smaller latency. When  $p_c$  is

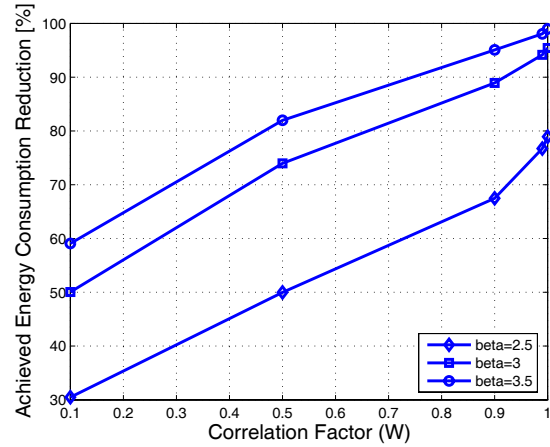


Fig. 13. The amount of energy consumption reduction vs. the amount of correlation ( $H = 5000$  m,  $R = 1000$  m).

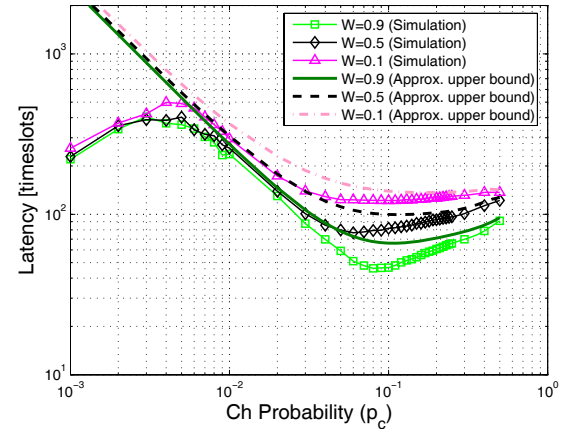


Fig. 14. The effect of correlation on system latency ( $H = 5000$  m,  $R = 1000$  m).

small, few clusters are formed, and communication is mainly performed within clusters. Therefore, the MPR capability of the data sink can not be fully utilized, and the TDMA delay becomes the dominant component of data collection latency. Thus, at small  $p_c$ 's, the latency becomes independent of  $W$ . The Latency behaves similarly for large values of  $p_c$ .

Considering Fig. 12 and Fig. 14 reveals the fact that correlation affects the energy consumption and the total delay almost in the same way. In both cases, a smaller  $W$  requires a larger cluster size (lower  $p_c$ ), and increases the minimum energy consumption and delay. This similarity can be attributed to the fact that correlation directly determines the size of the aggregated data, and the fact that the intra-cluster and reachback parts of energy consumption and latency for fixed packet lengths increase almost linearly with the number of cluster-members and the number of CHs, respectively.

Finally, combining the results for the energy consumption and latency, we conclude that in most cases there is a trade off between minimizing the overall delay and the energy consumption. Despite this tradeoff, for almost all parameter ranges, clustering reduces both latency and the total energy

consumption. To obtain a desired level of performance, the system designer can use the proposed analytical framework to tune  $p_c$  given that the other parameters of the system including  $H$ ,  $R$ ,  $\beta$ , and  $W$  are pre-specified.

## VII. CONCLUSION

In this paper, we have proposed a novel cluster-based data collection scheme for wireless sensor networks with direct sink access (CDC-DSA). Initiated by the data sink, clusters in CDC-DSA are formed in a distributed and random manner. We have used a simple MAC protocol and shown that to minimize the latency, the set of transmission probabilities can be obtained via dynamic programming. To simplify the design, we have proposed an alternate solution and proved that it is asymptotically optimal. Furthermore, we have shown that even for small networks the proposed solution performs almost optimally. We have developed our analytical framework based on the proposed MAC and the correlation between data samples. Our analysis has enabled us to obtain the optimal average cluster sizes and upper bounds for the data collection latency. It has been concluded that even when perfect aggregation is not possible, proper clustering can substantially decrease energy consumption and the latency. We have shown that, however, in general there is a tradeoff between minimizing energy consumption and minimizing latency. We have also studied the robustness of our protocol when the protocol does not have full knowledge of the number of contending CHs. We have concluded that a fixed estimation or an imperfect estimation provided by the clustering process suffices if small increases in the latency are acceptable.

## APPENDIX

In this appendix, we prove Proposition 2, which essentially states that the optimal policy asymptotically performs the same as the best stationary policy. The definition of  $n_0$  implies that to determine  $a_n$ , it suffices to specify  $\{a_i\}_{i=n-n_0}^{n-1}$  and  $p_n$ . Considering this fact, we have the following definition.

*Defintion 1:* A sequence  $\{o_i\}_{i=n}^{\infty}$  is called an optimal sequence for a given sequence  $\{g_i\}_{i=n-n_0}^{n-1}$  if the following holds assuming  $o_i = g_i$ ,  $n - n_0 \leq i \leq n - 1$ ,

$$o_i = \min_{0 \leq p_i \leq 1} \frac{1}{1 - t_0^i} \left( 1 + \sum_{k=1}^{n_0} t_k^i o_{n-k} \right), \quad i \geq n. \quad (25)$$

*Lemma 1:* Let  $\mathcal{L}_1$  be a line defined by  $\mathcal{L}_1 : x \mapsto \rho_1 x + \kappa_1$  such that  $l_i \triangleq \rho_1 i + \kappa_1 \leq a_i^*$  for  $n - n_0 \leq i \leq n - 1$ , where  $a_i^*$  is the optimal value for  $a_i$ , i.e. the solution of dynamic programming, and  $n$  is a given integer. Then, for any given sequence  $\{p_i\}_{i=n}^{\infty}$ , the following holds

$$a_i^l \leq a_i, \quad i \geq n, \quad (26)$$

where  $\{a_i^l\}_{i=n}^{\infty}$  and  $\{a_i\}_{i=n}^{\infty}$  are obtained by (12), and letting  $\{a_i\}_{i=n-n_0}^{n-1}$  equal  $\{l_i\}_{i=n-n_0}^{n-1}$  and  $\{a_i^*\}_{i=n-n_0}^{n-1}$ , respectively.

*Proof:* The proof is a direct result of the fact that in (12) all coefficients, i.e.  $t_{ni}$ 's, are non-negative real numbers and hence, the sequence  $\{a_i\}_{i=n}^{\infty}$  is a monotonic increasing function of  $\{a_i\}_{i=n-n_0}^{n-1}$  given any fixed sequence  $\{p_i\}_{i=n}^{\infty}$ . ■

*Lemma 2:* Let  $\{a_i^{*l}\}_{i=n}^{\infty}$  be the optimal sequence corresponding to  $\{l_i\}_{i=n-n_0}^{n-1}$ , and  $l_i$  be specified as in Lemma 1. Then

$$a_i^{*l} \leq a_i^*, \quad i \geq n. \quad (27)$$

*Proof:* Let  $\{p_i^*\}_{i=n}^{\infty}$  be the sequence which leads to the optimal sequence  $\{a_i^*\}_{i=n}^{\infty}$ . By Lemma 1, for such choice of  $p_i^*$ 's we have  $a_i^l \leq a_i^*$ , for  $i \geq n$ . By definition, the optimal sequence  $\{a_i^{*l}\}_{i=n}^{\infty}$  is componentwise smaller than  $\{a_i^l\}_{i=n}^{\infty}$ , and therefore,  $a_i^{*l} \leq a_i^*$  for  $i \geq n$ , as required. ■

*Lemma 3:* Let  $\epsilon > 0$  be arbitrarily small and  $\kappa$  chosen such that  $l_i = (\rho^* - \epsilon)i + \kappa \leq a_i^*$ , for  $n - n_0 \leq i \leq n - 1$ , where  $n$  is a sufficiently large number. Then,  $a_i^*$  satisfies the following:

- (a)  $(\rho^* - \epsilon)i + \kappa \leq a_i^*$ ,  $i \geq n$ .
- (b)  $\lim_{i \rightarrow \infty} \frac{a_i^*}{i\rho^*} \geq 1$ .

*Proof:*

- (a) Let  $a_i^l$ ,  $i \geq n$ , be obtained from (12), assuming  $a_i^l = l_i$ ,  $n - n_0 \leq i \leq n - 1$ . For any given  $p_n$ , we have

$$\begin{aligned} a_n^l &= \frac{1}{(1 - t_0^n)} \left( 1 + \sum_{i=1}^{n_0} t_i^n a_{n-i}^l \right) \\ &= \frac{1}{(1 - t_0^n)} \left( 1 + \sum_{i=1}^{n_0} t_i^n ((\rho^* - \epsilon)(n - i) + \kappa) \right) \\ &= \frac{1}{(1 - t_0^n)} \left( ((\rho^* - \epsilon)n + \kappa)(1 - t_0^n) \right. \\ &\quad \left. + (1 - (\rho^* - \epsilon)) \sum_{i=1}^{n_0} t_i^n i \right) \\ &= (\rho^* - \epsilon)n + \kappa + \frac{1}{(1 - t_0^n)} \left( 1 - (\rho^* - \epsilon) \sum_{i=1}^{n_0} t_i^n i \right). \end{aligned} \quad (28)$$

By the definition of  $\rho^*$ , for sufficiently large  $n$ , the last term in the above becomes non-negative. Thus,  $(\rho^* - \epsilon)n + \kappa \leq a_n^{*l}$ , and from Lemma 2, we obtain  $(\rho^* - \epsilon)n + \kappa \leq a_n^*$ . By induction,  $(\rho^* - \epsilon)i + \kappa \leq a_i^*$ , for  $i \geq n$ , as required.

- (b) Dividing both sides of the inequality in (a) by  $i\rho^*$ , and taking the limit as  $i \rightarrow \infty$ , we obtain  $1 - \frac{\epsilon}{\rho^*} \leq \lim_{i \rightarrow \infty} \frac{a_i^*}{i\rho^*}$ . The inequality in (b) immediately follows since  $\epsilon$  can be chosen arbitrarily small. ■

*Proof of Proposition 2:* Since  $a_n^*$  is the optimal solution, we have  $\lim_{n \rightarrow \infty} \frac{a_n^*}{a_n^{\xi^*}} \leq 1$ . Using Proposition 1 and Lemma 3, we can obtain the other direction of the inequality

$$\lim_{n \rightarrow \infty} \frac{a_n^*}{a_n^{\xi^*}} = \lim_{n \rightarrow \infty} \frac{n\rho^*}{a_n^{\xi^*}} \lim_{n \rightarrow \infty} \frac{a_n^*}{n\rho^*} \geq 1,$$

thus, proving the proposition.

## REFERENCES

- [1] G.J.Pottie and W. Kaiser, "Wireless integrated network sensors," *Communications of The ACM*, vol. 43, no. 5, pp. 892-900, May 2000.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, pp. 102-114, Aug. 2002.

- [3] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor network," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.
- [4] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks," in *Proc. of the First International Conference on Embedded Networked Sensor Systems*, Nov. 2003, pp. 193–204.
- [5] O. Younis and S. Fahmy, "Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach," in *Proc. IEEE INFOCOM*, Mar. 2004, pp. 629–640.
- [6] K. Romer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Commun. Mag.*, vol. 11, no. 6, pp. 54–61, Dec. 2004.
- [7] D. Niculescu, "Communication paradigms for sensor networks," *IEEE Commun. Mag.*, vol. 43, no. 3, pp. 116–122, Mar. 2005.
- [8] P. Venkatasubramanian, S. Adireddy, and L. Tong, "Sensor networks with mobile access: Optimal random access and coding," *IEEE J. Select. Areas Commun.: Special Issue on Sensor Networks*, vol. 22, no. 6, pp. 1058–1068, Aug. 2004.
- [9] V. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, "A minimum cost heterogeneous sensor network with a lifetime constraint," *IEEE Trans. Mobile Comput.*, vol. 4, no. 1, pp. 4–15, Jan./Feb. 2005.
- [10] G. Mergen, Q. Zhao, and L. Tong, "Sensor networks with mobile access: Energy and capacity considerations," *IEEE Trans. Commun.*, vol. 54, no. 11, pp. 2033–2044, Nov. 2006.
- [11] L. Tong, Q. Zhao, and S. Adireddy, "Sensor networks with mobile agents," in *Proc. IEEE MILCOM*, Oct. 2003, pp. 688–693.
- [12] S. Ghez, S. Verdu, and S. Schwartz, "Stability properties of slotted aloha with multipacket reception capability," *IEEE Trans. Automat. Contr.*, vol. 33, no. 7, pp. 640–649, July 1988.
- [13] Q. Zhao and L. Tong, "Quality-of-service specific information retrieval for densely deployed sensor networks," in *Proc. IEEE MILCOM*, Oct. 2003, pp. 591–596.
- [14] M. Lotfinezhad and B. Liang, "Effect of partially correlated data on clustering in wireless sensor networks," in *Proc. IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON'04)*, Oct. 2004, pp. 172–181.
- [15] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. IEEE INFOCOM*, vol. 3, Mar./Apr. 2003, pp. 1713–23.
- [16] S. J. Baek, G. Veciana, and X. Su, "Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation," *IEEE J. Select. Areas Commun.*, vol. 22, no. 6, pp. 1130–1140, Aug. 2004.
- [17] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Networking*, vol. 10, no. 4, pp. 477–486, Aug. 2002.
- [18] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang, "A twotier data dissemination model for largescale wireless sensor networks," in *Proc. of the 8th ACM Annual International Conference on Mobile Computing and Networking (MobiCom'02)*, Sept. 2002, pp. 148–159.
- [19] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling a three-tier architecture for sparse sensor networks," in *Proc. of the First IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003, pp. 30–41.
- [20] R. Rao and G. Kesidis, "Purposeful mobility for relaying and surveillance in mobile ad hoc sensor networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 3, pp. 225–231, July/Aug. 2004.
- [21] Z. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli, "Exploiting sink mobility for maximizing sensor networks lifetime," in *Proc. of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05)*, Jan. 2005.
- [22] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. Networking*, vol. 11, no. 1, pp. 2–16, Feb. 2003.
- [23] Q. Tian and E. Coyle, "A mac-layer retransmission algorithm designed for the physical-layer characteristics of clustered sensor networks," *IEEE Trans. Wireless Commun.*, vol. 5, no. 11, pp. 3153–3164, Nov. 2006.
- [24] M. Dong, L. Tong, and B. Sadler, "Source reconstruction via mobile agents in sensor networks: Throughput-distortion characteristics," in *Proc. IEEE MILCOM*, vol. 1, Oct. 2003, pp. 694–698.
- [25] Z. Yang and L. Tong, "Capacity of cooperative sensor networks with sensor errors," in *Proc. IEEE International Conference on Communications (ICC'05)*, Mar. 2005.
- [26] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, "Protocols for self-organization of a wireless sensor network," *IEEE Personal Commun. Mag.*, vol. 7, pp. 16–27, Oct. 2000.
- [27] Z. J. Haas and J. Deng, "Dual busy tone multiple access (DBTMA)—a multiple access control scheme for ad hoc networks," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 975–985, June 2002.
- [28] B. Liang and Z. Haas, "Virtual backbone generation and maintenance in ad hoc network mobility management," in *Proc. IEEE INFOCOM*, Mar. 2000, pp. 1293–1302.
- [29] R. Virrankoski and A. Savvides, "TASC: topology adaptive spatial clustering for sensor networks," in *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS'05)*, Nov. 2005, pp. 605–614.
- [30] S. Yoon and C. Shahabi, "Exploiting spatial correlation towards an energy efficient clustered aggregation technique (CAG)," in *Proc. IEEE International Conference on Communications (ICC'05)*, May 2005, pp. 3307–3313.
- [31] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "On network correlated data gathering," in *Proc. IEEE INFOCOM*, Mar. 2004, pp. 2571–2582.
- [32] R. Cristescu and B. Beferull-Lozano, "Lossy network correlated data gathering with high-resolution coding," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2817 – 2824, 2006.
- [33] Q. Zhao and L. Tong, "A dynamic queue protocol for multiaccess wireless networks with multipacket reception," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2221–2231, Nov. 2004.
- [34] J. Lehnert and M. Pursley, "Error probabilities for binary direct-sequence spread-spectrum communications with random signature sequences," *IEEE Trans. Commun.*, vol. 35, no. 1, pp. 87–98, Jan. 1987.
- [35] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. John Wiley, 1991.
- [36] A. Scaglione, "Routing and data compression in sensor networks: Stochastic models for sensor data that guarantee scalability," in *Proc. IEEE Intl. Symposium on Information Theory (ISIT'03)*, 2003.
- [37] T. Berger, *Rate Distortion Theory: A Mathematical Basis For Data Compression*, ser. Prentice-Hall series in information and system science. Englewood Cliffs, N.J. Prentice-Hall, 1971.
- [38] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. McGraw-Hill, 2002.
- [39] S. G. Foss and S. Zuyev, "On a voronoi aggregative process related to a bivariate poisson process," *Advances in Applied Probability*, vol. 28, no. 4, pp. 965–981, 1996.
- [40] J. Filar and K. Vrieze, *Competitive Markov Decision Processes*. New York, NY: Springer-Verlag, 1997.



**Mahdi Lotfinezhad** received the B.Sc. degree from the Sharif University of Technology in 2003 and the M.A.Sc. degree from the University of Toronto in 2004, both in electrical and computer engineering. He is currently a Ph.D. candidate in electrical and computer engineering at the University of Toronto. He has performed research in the area of wireless sensor networks with emphasis on joint sensor clustering and data aggregation. His current research focus is on complexity and stability properties of scheduling policies in wireless networks.



**Ben Liang** received honors simultaneous B.Sc. (valedictorian) and M.Sc. degrees in electrical engineering from Polytechnic University in Brooklyn, New York, in 1997 and the Ph.D. degree in electrical engineering with computer science minor from Cornell University in Ithaca, New York, in 2001. In the 2001 - 2002 academic year, he was a visiting lecturer and post-doctoral research associate at Cornell University. He joined the Department of Electrical and Computer Engineering at the University of Toronto in 2002, where he is now an Associate Professor.

His current research interests are in mobile networking and multimedia systems. He won an Intel Foundation Graduate Fellowship in 2000 toward the completion of his Ph.D. dissertation and an Early Researcher Award (ERA) given by the Ontario Ministry of Research and Innovation in 2007. He was a co-author of the Best Paper Award at the IFIP Networking Conference in 2005 and the Runner-up Best Paper Award at the International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks in 2006. He is an associate editor of *Wiley Security and Communication Networks* and serves on the organizational and technical committees of a number of conferences each year including ACM MobiCom, IEEE INFOCOM, IEEE MASS, and IEEE SECON. He is a senior member of IEEE and a member of ACM and Tau Beta Pi.



**Elvino S. Sousa** received his B.A.Sc. in engineering science, and the M.A.Sc. in Electrical Engineering from the University of Toronto in 1980 and 1982 respectively, and his Ph.D. in electrical engineering from the University of Southern California in 1985. Since 1986 he has been with the department of Electrical and Computer Engineering at the University of Toronto where he is now a Professor and the Jeff Skoll Professor in Computer Network Architecture. He has performed research in spread spectrum systems and CDMA since 1983. His current

interests are in the areas of high-speed CDMA systems, smart antenna systems, software radio, ad-hoc networks, and wireless system concepts for 4th generation networks. At the University of Toronto he is the director of the wireless lab, which has undertaken research in wireless systems for the past 20 years. He has been invited to give lectures and short courses on spread spectrum, CDMA, and wireless communications in a number of countries, and has been a consultant to industry and Governments internationally in the area of wireless systems. He was the technical program chair for PIMRC 95, and vice-technical program chair for Globecom '99, and has been involved in the technical program committee of numerous international conferences. He is a past chair of the IEEE Technical committee on Personal Communications. He has spent sabbatical leaves at Qualcomm and Sony CSL/ATL, where he was the holder of the Sony sabbatical chair. His current research interests are in the areas of autonomous infrastructure wireless networks, cognitive radio, multi-hop cellular networks, and self configurable wireless networks.