Analysis of Energy Efficiency of Compressive Sensing in Wireless Sensor Networks

Celalettin Karakus, Ali Cafer Gurbuz, and Bulent Tavli

Abstract—Improving the lifetime of wireless sensor networks (WSNs) is directly related to the energy efficiency of computation and communication operations in the sensor nodes. Compressive sensing (CS) theory suggests a new way of sensing the signal with a much lower number of linear measurements as compared to the conventional case provided that the underlying signal is sparse. This result has implications on WSN energy efficiency and prolonging network lifetime. In this paper, the effects of acquiring, processing, and communicating CS-based measurements on WSN lifetime are analyzed in comparison to conventional approaches. Energy dissipation models for both CS and conventional approaches are built and used to construct a mixed integer programming framework that jointly captures the energy costs for computation and communication for both CS and conventional approaches. Numerical analysis is performed by systematically sampling the parameter space (i.e., sparsity levels, network radius, and number of nodes). Our results show that CS prolongs network lifetime for sparse signals and is more advantageous for WSNs with a smaller coverage area.

Index Terms—Compressive sensing (CS), energy efficiency, mixed integer programming, network lifetime, wireless sensor networks (WSN).

I. INTRODUCTION

WIRELESS Sensor Networks (WSNs) are comprised of spatially distributed sensor nodes, where each node contains units for sensing, processing, and communicating data [1]. In general, sensor nodes are assumed to have limited processing power and highly constrained energy resources [2]. A typical WSN topology includes a base station - a powerful entity more capable than the ordinary sensor nodes with a significantly higher energy budget [3]. Ordinary sensor nodes transfer processed or raw sensed data to the base station, which performs the final information aggregation and extraction tasks [4].

In conventional signal processing techniques for true reconstruction at the base station, ordinary sensor nodes sample data at the Nyquist rate, generating raw measurements of the signal. Depending on the sophistication of the sensor, the signal can be transformed to a new domain where most of the signal energy can be represented by a small number of coefficients (i.e., the signal is compressible or sparse). Later these coefficients and their locations are encoded and then transmitted to the base station. Alternatively, each sensor node can also transmit its raw measurements to the base station without any processing. For example, in an image acquisition operation, the sensor first acquires raw data, which corresponds to measuring each pixel value. If the image is compressible in discrete cosine transform (DCT) space, the raw image can be transformed to the DCT domain [5]. In this way only a small number of DCT coefficients and their locations are saved. These coefficients constitute most of the energy in the image. The rest of the coefficients are discarded without deteriorating the perceived quality of the image significantly. Either the raw image pixels or the DCT coefficients may be transmitted depending on the selected technique.

Apart from these conventional techniques, the theory of Compressive Sensing (CS) [6], [7] proposes a novel signal acquisition and recovery method. Briefly, CS theory states that if a signal is sparse or compressible in a certain basis, then it can be reconstructed from a smaller number of linear measurements in comparison to the conventional case by solving an $\ell_1$ based convex optimization problem. The required number of measurements are linearly related to the underlying signal sparsity level. An example image reconstruction result is presented in [5]. Using CS for WSN applications, the sensor nodes can directly acquire a small number of measurements as linear projections of the raw signal and directly transmit these CS measurements to the base station without any further processing in the sensor node [8]–[12]. In this way, the signal can be acquired at its information rate and data is compressed while being sensed. This technique also eliminates the need to acquire data that is discarded after doing the transform coding. Although CS needs to transmit much less data compared to transmitting the whole raw data, it actually transmits more measurements as compared to the transform coding case. Hence, using a fair energy dissipation model (including both communication and computation energy costs), a comparison between conventional and CS based techniques can be performed to understand the conditions under which CS can improve energy efficiency, and enable longer lifetimes for WSNs.

Mixed Integer Programming (MIP) based analysis of communication networks is extremely useful for uncovering the fundamental performance limits [13]. Choosing an MIP based analysis method has a number of advantages. One of them is the abstraction from a specific protocol which enables us
to investigate energy cost in ideal conditions with optimal routing decisions. Secondly, due to global knowledge in the optimization problem solver, the results can be obtained in an efficient and consistent manner.

The novel contribution of our study is to characterize the tradeoffs in employing CS instead of conventional signal processing techniques in WSNs from energy efficiency and network lifetime perspectives. We developed a unified computation and communication energy model based on experimental characterizations of mica motes to account for different aspects of data acquisition, processing, and communication in sensor nodes that is capable of differentiating the energy dissipation of different sensing techniques. Based on this model, a novel MIP framework that jointly models the energy dissipation of both computation and communication operations in WSNs for varying sensing techniques is constructed.

The rest of this paper is organized as follows. In Section II, a brief overview of the literature on the use of CS theory in WSNs and on mathematical programming based analysis of WSNs is presented. Section III briefly outlines the mathematical background on compressive sensing that is used to develop the system model. In Section IV, a unified energy model that jointly captures the energy dissipation characteristics of WSN nodes for sensing, computation, and communication using available data on WSN hardware platforms. In Section V, we present our network model formulated as an MIP framework. Section VI presents the results of numerical analysis performed using the MIP framework by systematically exploring the parameter space. Conclusions are drawn in Section VII.

II. RELATED WORK

There are many studies on applications of CS theory on WSNs. In [14], novel models for joint sparsity in WSN applications is introduced and the benefits of distributed compressed sensing (e.g., signal recovery with fewer measurements, robustness, low complexity at sensor nodes) is demonstrated. In [9], several random routing methods for WSNs exploiting CS based measurement techniques are proposed. Analysis of the proposed methods in comparison to existing data gathering schemes is performed. In [10], the potential of CS based signal acquisition for low-complexity energy-efficient ECG compression on a wireless body sensor network mote (Shimmer) is investigated. The results of this study reveal that CS represents a competitive alternative to digital wavelet transform based ECG compression solutions. In [11], a CS based approach for monitoring environmental information using WSNs is presented. The proposed methods exploits the compressibility of the signal to reduce the amount of acquired data. In [12], a CS based data collection algorithm for WSNs is proposed. Theoretical analysis of the proposed algorithm shows that it can accurately recover data from a small amount of compressed data. In [15], a decentralized networking scheme that combines the concepts of random channel access and CS to achieve energy and bandwidth efficiency for underwater sensor networks is proposed. The concept of sufficient sensing probability is introduced to compensate for the random packet loss caused by collisions.

In [16], an algorithm employing compressive sensing in conjunction with particle swarm optimization to decrease the communication rate and to build up the data aggregation trees is proposed. It is shown through simulations that the proposed algorithm outperforms LEACH and Shortest-path routing in extending WSN lifetime. In [17], an optimization model for minimizing the network energy consumption through joint routing and compressed aggregation is developed. Both MIP-based and heuristic solutions are proposed. In [18], an approach utilizing the concepts of compressive sensing to minimize the number of packets to transmit in WSNs is proposed. Performance analysis using data sets gathered by a real-life deployment demonstrate that the approach helps finding an optimal tradeoff between the energy spent in transmission and data compression. In [19], an algorithm for compressive sensing in WSNs using rateless coding is proposed to keep the energy cost of inter-communications for generating projections. The algorithm is independent of routing algorithms or network topologies and provides the advantage of using non-uniform and unequal error protection codes. In [20], a modification of the canonical compressive sensing recovery is introduced to reduce the energy cost of event detection applications in WSNs. A practical implementation of the proposed scheme with energy constrained WSN nodes quantify the gains accrued through simulation and experimentation.

The literature on MIP based modeling and analysis of WSNs is extensive and has grown rapidly in recent years. Providing a comprehensive overview of the published research on modeling WSNs through MIP is beyond the scope of our work. We refer interested readers to the recent review papers on this topic [21], [22].

III. COMPRESSIVE SENSING THEORY

In conventional signal processing, a sensor acquires the signal at least at its Nyquist rate for proper reconstruction. Let's represent this acquired discrete signal as one dimensional vector $x \in \mathbb{R}^N$. Any vector in $\mathbb{R}^N$ can be represented as a linear combination of basis vectors $\{\psi_i\}_i^{N}$ as

$$x = \sum_{i=1}^{N} s_i \psi_i$$

where $\Psi$ is the basis matrix with $i^{th}$ column $\psi_i$. The signal $x$ is called $K$-sparse if only $K$ of the coefficients in transform domain vector $s$ is nonzero. The compressibility of most practical signals is the basic point for transform coding. A wireless sensor node, depending on its sophisticated, can either transmit all $N$ measurements without any processing or it can transform the signal to a new domain where it can be represented with $K \ll N$ coefficients. In transform coding, the full signal $x \in \mathbb{R}^N$ is acquired; all transform coefficients are calculated by $s = \Psi^T x$; the largest $K$ coefficients are located and the rest are discarded. Finally, only the largest $K$ coefficients and their locations are encoded and transmitted.

Although transform coding decreases the amount of data communicated, several fundamental points should be mentioned.
1) Although only \( K \ll N \) coefficients are sought, all \( N \) measurements, which may be large, are needed to be acquired.
2) All the transform coding coefficients are computed even though only \( K \) of them will be used and the rest are discarded.
3) Locations of the coefficients must also be encoded introducing an extra overhead.

The compressive sensing (CS) theory [6], [7] addresses these inefficiencies by compressing while acquiring data at its information rate. CS takes nontraditional linear measurements, \( y = \Phi x \), in the form of randomized projections. A signal \( x \), which is \( K \) sparse in \( \Psi \) can be reconstructed from \( M = O(K \log N) \) compressive measurements, where \( M \) is the number of required compressive measurements. This reconstruction requires solving a convex optimization problem of the following form

\[
\arg \min \| s \|_1 \quad s.t. \quad y = \Phi \Psi s
\]  

(2)

It is also shown in [23] and [24] that stable recovery of \( s \) can be done by relaxing the optimization problem in Equation 2 for noisy measurements. In CS, much less number of measurements \( M \) compared to the data acquisition number \( N \) is taken, however \( M \) is larger than the sparsity level \( K \) as \( M = O(K \log N) \) and no transform coding should be done at the wireless sensor. Instead, \( M \) compressed measurements can be transmitted to the base station and the reconstruction can be done there. The reconstruction in CS is done by solving an optimization problem, such as in Equation 2. This optimization problem is convex and can be solved using linear programming and the global optimal solutions can be achieved. The computational complexity of this solution is greater than that of transform coding, but it is done at the base station, which we assume to have a significantly higher energy budget than the ordinary sensor nodes. The base station must know the measurement matrix \( \Phi \) of the wireless sensor to solve the optimization problem in Equation 2. The measurement matrix \( \Phi \) does not change through the lifetime of the wireless sensors, hence, the sensor nodes can be preloaded with this data before deployment.\(^1\) Alternatively, the measurement matrix can be transmitted to the sensor nodes by the base station; however, such data dissemination is not frequent, and thus, does not lead to any significant energy consumption.

IV. ENERGY MODEL

In this section, we develop an energy dissipation model for three approaches\(^2\):

1) Data Acquisition and No Processing (DANP) approach
2) Data Acquisition and Transform Coding (DATC) approach
3) Data Acquisition and Compressive Sensing (DACS) approach

\(^1\)In our analysis, we adopted this approach.
\(^2\)In the rest of the paper, DANP and DATC are referred to as conventional approaches and DATC is referred to as the CS-based approach.

Energy dissipation in a typical WSN node can be categorized into two groups: (i) energy dissipation due to computation – \( E_{CM} \) and (ii) energy dissipation due to communication – \( E_{COM} \). We used the energy dissipation characteristics of the Mica2 platform to determine the energy dissipation model. Mica2 platform consists of an Atmel Atmega 128L processor and Chipcon CC1000 radio. Both of them have very well characterized energy dissipation properties. We use the communication energy dissipation model for Mica2 motes equipped with CC1000 radios presented in [25]. Transmission ranges and corresponding energy dissipations for this model are presented in Table I. Energy dissipation for transmitting one bit of data at power level \( l \) is denoted as \( E^{\text{tx}}_{l,0} \) and the maximum transmission range at power level \( l \) is denoted as \( R^{\text{tx}}_{l,\text{max}} \). If the distance between node-\( i \) and node-\( j \) is larger than \( R^{\text{tx}}_{l,\text{max}} \) (i.e., \( d_{ij} > R^{\text{tx}}_{l,\text{max}} \)) then they cannot communicate using power level \( l \). Energy dissipation for receiving one bit of data is constant and denoted as \( E_{r,0} \). Each data packet has a header length of 168 bits and the maximum packet size is 2040 bits, thus, the maximum data payload per packet is 1872 bits [26]. Acknowledgement packet length is 160 bits (\( L_A = 160 \)).

Energy dissipation for computation is comprised of three main components:

1) data acquisition energy dissipation – \( E_{ACQ} \)
2) background energy dissipation – \( E_{BCK} \)
3) energy dissipation for processing – \( E_{SP} \)

<table>
<thead>
<tr>
<th>Power Level (l)</th>
<th>Transmission Energy ( (E^{\text{tx}}_{l,0}) )</th>
<th>Range ( (R^{\text{tx}}_{l,\text{max}}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>l-1</td>
<td>0.672</td>
<td>19.30</td>
</tr>
<tr>
<td>l-2</td>
<td>0.688</td>
<td>20.46</td>
</tr>
<tr>
<td>l-3</td>
<td>0.702</td>
<td>21.09</td>
</tr>
<tr>
<td>l-4</td>
<td>0.706</td>
<td>22.09</td>
</tr>
<tr>
<td>l-5</td>
<td>0.711</td>
<td>24.38</td>
</tr>
<tr>
<td>l-6</td>
<td>0.724</td>
<td>25.84</td>
</tr>
<tr>
<td>l-7</td>
<td>0.727</td>
<td>27.39</td>
</tr>
<tr>
<td>l-8</td>
<td>0.742</td>
<td>29.03</td>
</tr>
<tr>
<td>l-9</td>
<td>0.758</td>
<td>30.78</td>
</tr>
<tr>
<td>l-10</td>
<td>0.773</td>
<td>32.62</td>
</tr>
<tr>
<td>l-11</td>
<td>0.789</td>
<td>34.58</td>
</tr>
<tr>
<td>l-12</td>
<td>0.813</td>
<td>36.66</td>
</tr>
<tr>
<td>l-13</td>
<td>0.828</td>
<td>38.86</td>
</tr>
<tr>
<td>l-14</td>
<td>0.844</td>
<td>41.19</td>
</tr>
<tr>
<td>l-15</td>
<td>0.867</td>
<td>43.67</td>
</tr>
<tr>
<td>l-16</td>
<td>1.078</td>
<td>46.29</td>
</tr>
<tr>
<td>l-17</td>
<td>1.135</td>
<td>49.07</td>
</tr>
<tr>
<td>l-18</td>
<td>1.135</td>
<td>52.01</td>
</tr>
<tr>
<td>l-19</td>
<td>1.180</td>
<td>55.13</td>
</tr>
<tr>
<td>l-20</td>
<td>1.234</td>
<td>58.44</td>
</tr>
<tr>
<td>l-21</td>
<td>1.313</td>
<td>61.95</td>
</tr>
<tr>
<td>l-22</td>
<td>1.344</td>
<td>65.67</td>
</tr>
<tr>
<td>l-23</td>
<td>1.445</td>
<td>69.61</td>
</tr>
<tr>
<td>l-24</td>
<td>1.500</td>
<td>73.79</td>
</tr>
<tr>
<td>l-25</td>
<td>1.664</td>
<td>78.22</td>
</tr>
<tr>
<td>l-26</td>
<td>1.984</td>
<td>82.92</td>
</tr>
<tr>
<td>l-27</td>
<td>2.358</td>
<td>100.00</td>
</tr>
</tbody>
</table>
Therefore, computation energy dissipation can be expressed as a sum

\[ E_{\text{CMP}} = E_{\text{ACQ}} + E_{\text{BCK}} + E_{\text{SP}} \]  

(3)

Power consumption for sensing (including the power consumption of both the CPU and the sensor board) in the Mica2 motes is measured as \( P_{\text{ACQ}} = 15.01 \text{mW} \) [28]. Acquisition of an \( N \)-byte raw signal requires \( N \) CPU operations. At an operation frequency of 7.4 MHz the Atmega 128L can execute 7.4 Machine Instructions Per Second (MIPS) \(^3\). [27]. Hence, the energy dissipation for acquiring an \( N \)-byte signal is obtained as follows:

\[ E_{\text{ACQ}} = N P_{\text{ACQ}} D_{\text{SP}} \]  

(4)

Note that most of the instruction set of Atmega 128L are executed in a single CPU clock cycle [27]. Furthermore, measurements presented in [29] show that CPU access to various peripherals does not draw more current than any other CPU instruction.

If the raw data is to be communicated without any signal processing operation (e.g., the sensors might not be able to do the transform coding operations by themselves) then the energy dissipation for computation is for signal acquisition only as expressed in Equation 5.

\[ E_{\text{CMP-DACP}} = E_{\text{ACQ}} \]  

(5)

Background energy dissipation does not vary for different signal processing techniques. The energy dissipation of various sources necessary to operate the platform is obtained by multiplying the power for running the CPU in the idle mode \( P_{\text{BCK}} = 9.6 \text{mW} \) [28] by the total CPU utilization time for signal processing \( (D_{\text{SP}}) \), as shown in Equation 6

\[ E_{\text{BCK}} = P_{\text{BCK}} D_{\text{SP}}. \]  

(6)

The number of operations for a particular signal processing task determines the amount of CPU time utilized. In other words, the background energy dissipation is scaled with the duration of the signal processing operations.

In DATC, a transformation using a transform basis is obtained for signal decomposition, yielding \( N \) transform coefficients. To obtain a \( K \)-sparse approximation of the signal, the \( K \) largest values with corresponding locations should be found. Locations and values of the \( K \) largest coefficients represent the signal. In the transform coding process, multiplication with a transform basis requires \( N^2 \) additions and \( N^2 \) multiplications. Sorting and selecting large coefficients requires \( \text{Nlog}(N) \) comparisons. Thus, the computation energy dissipation in the sensor CPU for DATC is expressed as

\[ E_{\text{SP-DATC}} = N e_{\text{mrd}} + N^2 (e_{\text{add}} + e_{\text{mul}}) + N \text{log}N (e_{\text{cmp}} + e_{\text{fsl}}) + 2K e_{\text{mur}}, \]  

(7)

where \( e_{\text{add}} \) (3.30 nJ), \( e_{\text{mul}} \) (9.90 nJ), \( e_{\text{cmp}} \) (3.30 nJ), \( e_{\text{fsl}} \) (3.30 nJ), \( e_{\text{mrd}} \) (0.26 nJ), and \( e_{\text{mur}} \) (4.30 nJ) are the energy dissipation values for addition, multiplication, comparison, shift, read, and write operations (per byte) in the CPU, respectively [26], [30]. The total number of operations involved in the process is denoted by \( O_{\text{SP-DATC}} \), which has a value of \( 2N^2 + 2N \text{log}(N) + N + 2K \) for this process. Hence, the total time required for this operation is \( D_{\text{SP-DATC}} = O_{\text{SP-DATC}} D_{\text{OP}} \).\(^4\)

The main idea behind compressive sensing is to compressively sense and acquire the signal at its information rate. Hence, the goal is to acquire \( M \) random projections of the analog signal, instead of measuring \( N \gg M \) measurements. Although there is a considerable amount of work on the development of CS-based sensing hardware, which acquires measurements as linear projections [31]–[33], these efforts are mainly at experimental stage. Moreover, complete energy models for such hardware platforms are not available. Hence, the CS energy model assumes an \( N \) dimensional signal acquired as in the conventional case but the \( M \) compressive measurements are generated using a multiplication process with a random \( M \times N \) Bernoulli (random \( \pm 1 \)) matrix within our comparison platform. Since the matrix contains only the values of \( \pm 1 \), it requires \( MN \) additions in terms of computation cost in addition to the read and write operations. The computation cost for DACS is

\[ E_{\text{SP-DACS}} = N e_{\text{mrd}} + MN e_{\text{add}} + M e_{\text{mur}}. \]  

(8)

The total number of operations for DACS is \( O_{\text{SP-DACS}} = MN + N + M \) and the time required for DACS is \( D_{\text{SP-DACS}} = O_{\text{SP-DACS}} D_{\text{OP}} \).

V. NETWORK MODEL

In this section, we formulate data processing and routing in the network as an MIP framework using the energy model presented in Section IV. We apply our models to conventional and compressive sensing based processing techniques.

In the MIP framework, we assume that there is a single base station and multiple sensor nodes (total number of nodes in the network is \( \varsigma \)). The network topology is represented by a connected graph \( G = (V, A) \). \( V \) is the set of all nodes, including the base station (node-0). We also define set \( W \), which includes all nodes except the base station (i.e., \( W = V \setminus \{0\} \)). \( A = \{(i, j) : i \in V, j \in V - i, d_{ij} \leq R^{1-27}_{\text{max}} \} \) is the ordered set of arcs. The total number of data packets transmitted by node-\( i \) to node-\( j \) is represented as \( f_{ij} \). The total number of acknowledgement packets transmitted in response to data packets are represented as \( g_{ij} \). Note that the definition of \( A \) implies that no node sends data to itself or to a node that is separated from it beyond the maximum transmission range \( R^{1-27}_{\text{max}} \).

Although there is no data flow from the base station to any sensor node\(^5\), the base station transmit acknowledgement packets, and, hence the arcs from the base station to the sensor nodes are included in the set of arcs. We assume that all nodes are roughly time synchronized and time is organized

\(^3\)The duration of instruction execution is \( D_{\text{OP}} = \frac{1}{7.4 \times 10^6} \).

\(^4\)Symbols and acronyms used in the paper and their descriptions are presented in the appendix (Table II).

\(^5\)It is possible to use a CSMA/CA type MAC protocol like IEEE 802.15.4 in beacon enabled mode for low energy dissipation, where, beacon packets are periodically transmitted by the base station to synchronize sensor nodes. Such an approach necessitates flow of beacon packets from the base station to the sensor nodes.
generated in the network is $H \times \sum_{i \in W} s_i$). Equation 12 states that the number of acknowledgement packets on arc $(j, i)$ is equal to the number of data packets on arc $(i, j)$. Equation 13 is used to guarantee that there are no data packets flowing out of the base station. Equation 14 states that for all nodes except the base station energy dissipation for communication and computation is bounded by the energy stored in batteries ($\varrho$). In the analysis, we use $\varrho = 25$ KJ (energy provided by two AA batteries) [29]. The optimal transmission energy value is found using Equation 16. For instance, for $d_{ij} = 20$ m, since $19.30 \text{ m} < d_{ij} \leq 20.46 \text{ m}$, node-i uses power level 2 to transmit data on its link to node-j (i.e., $E^{\text{opt}}_{ix,ij} = 0.688 \mu J$). Both the number of data bytes generated at each round and computation energy dissipation at each round are determined by the data representation approach selected. Lifetime of a network ends when the first node exhausts its energy. However, this definition should not be misinterpreted -- when we examine the framework carefully it can be seen that to maximize the minimum lifetime, all nodes are forced to dissipate their energies in a balanced fashion, hence, sensor nodes in the network deplete their battery energies simultaneously.

$$E^{\text{opt}}_{tx,ij} = \begin{cases} E^{i-1}_{tx} & \text{if } d_{ij} \leq R_{\text{max}}^{l-1} \\ \infty & \text{else if } d_{ij} > R_{\text{max}}^{l-1} \\ E^{i+1}_{tx} & \text{else if } R_{\text{max}}^{l} < d_{ij} \leq R_{\text{max}}^{l+1} \end{cases}$$

(9)

To take channel bandwidth limitations into consideration in a broadcast medium, we need to make sure that the bandwidth required to transmit and receive at each node is limited by the channel bandwidth. Such a constraint should take the shared capacity into consideration. We refer to the flows around node-i (both data and acknowledgement), which are not flowing into or flowing out of node-i and affect the available bandwidth to node-i, as interfering flows. Equation 15 guarantees that for all nodes including the base station the aggregate rate of incoming flows, outgoing flows, and interfering flows is upper bounded by the channel bandwidth. This constraint is a modified version of the sufficient condition given in [34]. The interference function ($I_{jl}^i$) is presented in Equation 17. If node-i is in the interference region of the transmission from node-j to node-l, then the value of interference function for node-i ($I_{jl}^i$) is unity, otherwise it is zero. Generally speaking, interference range is equal to or greater than transmission range (i.e., $\gamma \geq 1$). This means depending on the value of $\gamma$, node-j’s transmission to node-l can interfere with node-i even if the distance between node-j and node-l is less than the distance between node-j and node-i.

$$I_{jl}^i = \begin{cases} 1 & \text{if } \gamma d_{jl} \geq d_{ji} \forall j \in V \setminus \{i\}, \forall l \in V \setminus \{i, j\} \\ 0 & \text{otherwise} \end{cases}$$

(10)

WSNs are assumed to be consisting of stationary sensor nodes, thus, topology discovery, route creation, and other initialization operations (e.g., measurement matrix dissemination) are one-time operations -- for a substantial amount of time these functions are not repeated. If the network reorganization period is long enough, the energy costs of these operations constitute a small fraction (less than 1%) of the total network energy dissipation [35]. Hence, routing overhead can be neglected in stationary WSNs without leading to significant
underestimation of total energy dissipation. We opt to keep our model as simple as possible to eliminate the shadowing effects of implementation details not specifically related to the concept under investigation, *per se*.

VI. Analysis

In this section, we first perform numerical analysis with various sparsity levels to determine the required number of measurements for CS to reconstruct the acquired signal. The l1-magic packet [36] is used to solve the signal recovery problem. It is a collection of MATLAB routines for solving the convex optimization programs central to compressive sampling. We first use these results together the developed computation and communication energy dissipation models to investigate the effects of DANP, DATC, and DACS on network lifetime without considering data routing (i.e., communication and computation energy dissipation characteristics for a single sensor node is investigated). Later, we systematically explore and compare the aforementioned processing approaches’ effects on WSN lifetime by sampling the parameter space through the constructed MIP model, which can model both computation and communication energy dissipation terms within a unified framework.

We use the General Algebraic Modeling System (GAMS) [37] for numerical analysis of MIP models. GAMS is a high-level modeling system for mathematical programming and optimization. It consists of a language compiler and integrated high-performance solvers. GAMS is used widely for complex, large scale modeling applications.

To determine the energy consumption of a wireless node employing CS for a given signal of dimension \( N \) and sparsity level \( K \), the required number of measurements \( M \) for correct reconstruction is needed. CS theory defines this measurement number \( M \) as in the order of \( K \log N \) [7]. In the literature, the required number of measurements for correct reconstruction with CS is a studied topic and phase transition curves explaining these relations are obtained [38]. For the sake of completeness, we perform numerical analysis to determine \( M \) as a function of \( K \) and \( N \). A signal of length \( N = 512 \) is taken with sparsity levels \( K \) varying between 20 to 200. For each sparsity level \( K \), \( M \) compressive measurements are produced using a random Bernoulli/Rademacher (random ±1) measurement matrix of dimension \( M \times N \). Measurement numbers between 20 and 512 are tested and the signals are reconstructed using the \( l_1 \) minimization problem defined in Equation 2. These numerical experiments are run 500 times with independent random sparse signal and measurement matrix selections and the number of correct reconstructions are counted. Figure 2 shows the correct reconstruction ratio as a function of number of measurements for different sparsity levels.

The required number of measurements for each sparsity level is estimated from the numerical experiments as \( M \approx 1.5K\log N \) which is consistent with results in the literature. This relation is also tested with different signal dimensions \( N \) and successful recoveries for all cases are observed. This level of \( M \) is sufficient to recover a \( K \) sparse signal perfectly for all sparsity levels \( K \) used with CS. Hence, in the rest of our analysis, \( M = 1.5K\log N \) is used for the CS energy model computations.

To have a basic understanding of the tradeoffs involved in communication and computation without the shadowing effects of routing, first, the energy dissipation of a single wireless sensor node for acquisition, processing, and communication of \( N = 1024 \) bytes of data as a function of level of sparsity is analyzed. The DANP, DATC, and DACS approaches as detailed in Section IV are compared. Two transmission ranges of 50 m and 100 m are utilized, where the sensor node transmits data by using \( l-18 \) and \( l-27 \), respectively. Communication energy dissipation is due to data transmission and acknowledgement reception. Figure 3 presents energy consumption as a function of sparsity ratio \( K/N \).

We observe that the energy consumption levels for DACS and DATC increases with the sparsity level of the signal, while...
the energy consumption level of DANP stays constant since it transmits all the acquired $N$ data samples independent of the signal sparsity. Another observation is that the energy consumption for DACS is lower as compared to both DATC and DANP, provided that the signal is sparse enough (i.e., $K/N < 0.15$) for both ranges. On the other hand, for non-sparse signals, using CS is not advantageous. The energy consumption level of DATC is higher when compared to DACS due to the transform coding operations needed to be completed before transmitting. However, due to the higher communication energy dissipation of DACS, for transmissions over longer distances, DATC has better energy efficiency. To investigate the effects of different approaches on network lifetime, we use the MIP model introduced in Section V. We use a disk topology with radius $R_{net}$. Sensor nodes are randomly deployed and uniformly distributed on the network area. The base station is at the center of the disc. Each problem is solved for 200 random topologies and the average network lifetimes are obtained. Normalized network lifetime results for varying levels of sensor node number and network radius for sparsity levels of $K/N = 0.05, 0.10, 0.15,$ and $0.20$ are given. Normalization is achieved by dividing all data points in a figure by the largest value.

Figure 4 presents normalized network lifetime as a function of the number of nodes when the network radius is fixed to 100 meters for different sparsity levels. DACS has the highest network lifetime throughout the whole parameter space. Especially for sparser signals (e.g., $K/N = 0.05$) network lifetimes obtained with DACS is much larger than the lifetimes obtained with other approaches (e.g., DACS approach can result in a lifetime improvement of up to 4 times over DATC approach for $\zeta = 50$ and $K/N = 0.05$). However, as the sparsity level increases the difference between the network lifetimes obtained with CS and with other approaches decreases (e.g., network lifetimes of DACS and DANP are within 15% neighborhood of each other for $K/N = 0.20$). Network lifetime increases for all approaches with increasing number of nodes, since increasing the number of nodes for a fixed network radius $R_{net}$ increases the node density, which creates more paths towards the base station (i.e., number of node
of neighbor nodes increase). Larger network lifetimes can be obtained with richer routing options available in strongly connected networks. Furthermore, increasing the node density decreases the average hop distance. Note that transmission energy dissipation increases as the distance between the transmitter and the receiver increases. In summary, Figure 4 shows that:

1) WSN lifetime is significantly prolonged by DACS in comparison to conventional approaches (DATC and DANP) for all values of the number of nodes in the network. Efficiency of compressive sensing in reducing computation energy dissipation is the key factor in superior energy efficiency of DACS.

2) Lifetime gains obtained by DACS is higher for lower $K/N$ because, as characterized in Figure 3, energy dissipation for DACS increase as the ratio $K/N$ increase.

Figure 5 presents normalized network lifetime as a function of network radius for varying sparsity levels ($\zeta = 50$). For all values of $K/N$ except $K/N = 0.20$, DACS network lifetime is larger than the lifetimes obtained with other approaches provided that $R_{net} \leq 150$ m (e.g., DACS approach can result in a lifetime improvement of up to 4 times over DATC and DANP approaches for $R_{net} = 100$ m and $K/N = 0.05$). DATC outperforms DACS for larger values of network radius (i.e., $R_{net} \geq 250$ m for $K/N = 0.05$, $R_{net} \geq 200$ m for $K/N = 0.10$, $R_{net} \geq 150$ m for $K/N = 0.15$, and $R_{net} \geq 150$ m for $K/N = 0.20$) with respect to network lifetime. DANP has longer network lifetime values for smaller network radii and high $K/N$ ($R_{net} \leq 50$ m for $K/N = 0.20$).

Network lifetime decreases as $R_{net}$ increases because the average hop distance increases, which results in increased communication cost. Furthermore, increasing network area while keeping the number of nodes constant leads to a decrease in the number of neighbors per node. This limits the energy balancing capabilities of the network. In summary, Figure 5 shows that DACS provides longer network lifetimes in denser networks with lower $K/N$ because in such circumstances lower energy dissipation of DACS on computation can compensate its higher communication energy dissipation resulting in lower overall energy dissipation.

Fig. 5. Normalized network lifetime as a function of $R_{net}$ ($\zeta = 50$). (a) $K/N = 0.05$. (b) $K/N = 0.10$. (c) $K/N = 0.15$. (d) $K/N = 0.20$. 
VII. Conclusion

In this study, the energy dissipation characteristics of WSNs utilizing the concepts of compressive sensing is investigated and compared to two other well known conventional approaches (DANP and DATC). A model to quantify the energy dissipation in sensor nodes due to data acquisition, computation, and communication for the compared methods is developed using the measured characteristics of the Mica sensor network platform. By using the created energy dissipation models an MIP framework is built. The MIP framework models both computation and communication aspects within a unified framework. A systematic exploration of the parameter space, including sparsity level, node density, and network size, to characterize the energy dissipation and network lifetime performances of CS-based (DACS) and conventional (DANP and DATC) approaches is performed. Our results show that compressive sensing prolongs network lifetime significantly in comparison to conventional approaches provided that the acquired signals are highly sparse (e.g., \( K/N \leq 0.10 \)) and node density in the network is not too low (e.g., \( R_{\text{net}} \leq 150 \text{ m} \) and \( \zeta = 50 \)).

APPENDIX

<table>
<thead>
<tr>
<th>Sym/acy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x )</td>
<td>Vector representing acquired signal</td>
</tr>
<tr>
<td>( \Psi )</td>
<td>Basis vector in ( R^N )</td>
</tr>
<tr>
<td>( \Psi )</td>
<td>Basis matrix</td>
</tr>
<tr>
<td>( s )</td>
<td>Transform domain vector</td>
</tr>
<tr>
<td>( N )</td>
<td>Total number of unknowns</td>
</tr>
<tr>
<td>( M )</td>
<td>Total number of measurements in CS</td>
</tr>
<tr>
<td>( K )</td>
<td>Number of nonzero coefficients</td>
</tr>
<tr>
<td>( y )</td>
<td>CS measurements in ( R^M )</td>
</tr>
<tr>
<td>( \Phi )</td>
<td>CS measurement matrix in ( R^{M \times N} )</td>
</tr>
<tr>
<td>( E_{\text{CMP}} )</td>
<td>Computation energy dissipation</td>
</tr>
<tr>
<td>( E_{\text{COM}} )</td>
<td>Communication energy dissipation</td>
</tr>
<tr>
<td>( E_{\text{TX}}^{ij} )</td>
<td>Transmission energy dissipation at level ( l )</td>
</tr>
<tr>
<td>( R_{\text{max}}^{ij} )</td>
<td>Maximum transmission range at level ( l )</td>
</tr>
<tr>
<td>( d_{ij} )</td>
<td>Distance between node-i and node-j</td>
</tr>
<tr>
<td>( E_{\text{RX}} )</td>
<td>Reception energy dissipation (0.923( \mu )J/bit)</td>
</tr>
<tr>
<td>( \zeta )</td>
<td>Channel bandwidth (38.4 KHz/s)</td>
</tr>
<tr>
<td>( L_{P} )</td>
<td>Actual data packet length</td>
</tr>
<tr>
<td>( L_{A} )</td>
<td>ACK packet length (160 b)</td>
</tr>
<tr>
<td>( E_{\text{ACQ}} )</td>
<td>Data acquisition energy dissipation</td>
</tr>
<tr>
<td>( E_{\text{BCK}} )</td>
<td>Background energy dissipation</td>
</tr>
<tr>
<td>( E_{\text{SP}} )</td>
<td>Signal processing energy dissipation</td>
</tr>
<tr>
<td>( E_{\text{PACQ}} )</td>
<td>Sensing power consumption (15.01 mW)</td>
</tr>
<tr>
<td>( D_{\text{SP}} )</td>
<td>Instruction execution duration (0.14 ( \mu )s)</td>
</tr>
<tr>
<td>( E_{\text{CMP},\text{DANP}} )</td>
<td>Computation energy dissipation for DANP</td>
</tr>
<tr>
<td>( E_{\text{SP}} )</td>
<td>Signal processing CPU utilization time</td>
</tr>
<tr>
<td>( E_{\text{SP},\text{DATC}} )</td>
<td>Computation energy dissipation for DATC</td>
</tr>
<tr>
<td>( E_{\text{add}} )</td>
<td>Energy dissipation for addition (3.30 nJ)</td>
</tr>
<tr>
<td>( E_{\text{mul}} )</td>
<td>Multiplication energy dissipation (9.90 nJ)</td>
</tr>
<tr>
<td>( E_{\text{cmp}} )</td>
<td>Comparison energy dissipation (3.30 nJ)</td>
</tr>
</tbody>
</table>

REFERENCES


Celalettin Karakus received the B.S. degree in electrical and electronics engineering from the TOBB University of Economics and Technology, Ankara, Turkey, in 2010. He is currently pursuing the M.S. degree under the supervision of Dr. Tavli and Dr. Gurbuz.

He was a Research and Teaching Assistant with the Electrical and Electronics Engineering Department from 2010 to 2011. Since then, he has been with Roteksan Missiles Industries Inc., Ankara, where he is currently an Avionics System Design and Test Engineer. His current research interests include signal processing, compressive sensing, and wireless sensor networks.

Ali Cafer Gurbuz received the B.S. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2003, and the M.S. and Ph.D. degrees from the Georgia Institute of Technology (Georgia Tech), Atlanta, GA, USA, in 2005 and 2008, respectively, both in electrical and computer engineering.

He participated in multimodal landmine detection system research as a Graduate Research Assistant from 2003 to 2008 and from 2008 to 2009, as Post-Doctoral Fellow, with Georgia Tech. He is currently an Assistant Professor with the Department of Electrical and Electronics Engineering, TOBB University of Economics and Technology, Ankara. His current research interests include compressive sensing applications, ground penetrating radar, array signal processing, remote sensing, and imaging.

Bulent Tavli received the B.S. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 1996, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Rochester, Rochester, NY, USA, in 2001 and 2005.

He is currently an Associate Professor with the Department of Electrical and Electronics Engineering, TOBB University of Economics and Technology, Ankara, Turkey. His current research interests include telecommunications, networking, signal processing, and embedded systems.