Abstract—In this paper, a compressive sensing (CS) approach is proposed for sparse binary signals’ compression and reconstruction based on analog fountain codes (AFCs). In the proposed scheme, referred to as the analog fountain compressive sensing (AFCS), each measurement is generated from a linear combination of \( L \) randomly selected binary signal elements with real weight coefficients. The weight coefficients are chosen from a finite weight set and \( L \), called measurement degree, is obtained based on a predefined degree distribution function. We propose a simple verification based reconstruction algorithm for the AFCS in the noiseless case. The proposed verification based decoder is analyzed through SUM-OR tree analytical approach and an optimization problem is formulated to find the optimum measurement degree to minimize the number of measurements required for the reconstruction of binary sparse signals. We show that in the AFCS, the number of required measurements is of \( \mathcal{O}(n \log(1 - k/n)) \), where \( n \) is the signal length and \( k \) is the signal sparsity level. We then consider the signal reconstruction of AFCS in the presence of additive white Gaussian noise (AWGN) and the standard message passing decoder is then used for the signal recovery. Simulation results show that the AFCS can perfectly recover all non-zero elements of the sparse binary signal with a significantly reduced number of measurements, compared to the conventional binary CS and \( \ell_1 \)-minimization approaches in a wide range of signal to noise ratios (SNRs). Finally, we show a practical application of the AFCS for the sparse event detection in wireless sensor networks (WSNs), where the sensors’ readings can be treated as measurements from the CS point of view.

Index Terms—Analog fountain codes, compressive sensing, message passing, wireless sensor network.

I. INTRODUCTION

COMPRESSIVE SENSING (CS) has drawn a lot of interests in recent years as a revolutionary signal sampling paradigm. The main idea behind CS is the possibility to reconstruct a sparse signal from an under-determined system of linear equations through a computationally efficient convex programming algorithm. Let us consider the reconstruction problem of a sparse signal \( x \in \mathbb{R}^n \) from the linear measurements \( y \in \mathbb{R}^m \), where \( m < n \), \( y = Gx \), and \( G \in \mathbb{R}^{m \times n} \) is the measurement matrix, modeling the measurement (sensing) process. We assume that the signal \( x \) is \( k \)-sparse, which means that exactly \( k \ll n \) elements of \( x \) are nonzero \([1]\), but we do not know the locations of the nonzero entries of \( x \). Due to the sparsity of \( x \), one could try to compute \( x \) by solving the following optimization problem:

\[
\min_x ||x||_0 \quad \text{s.t.} \quad Gx = y. \tag{1}
\]

However, solving (1) is an NP-hard problem and thus practically not feasible \([2]\). Instead, a convex relaxation problem can be formulated as follows, which can be solved efficiently via linear or quadratic programming techniques.

\[
\min_x ||x||_1 \quad \text{s.t.} \quad Gx = y. \tag{2}
\]

As shown in \([2–5]\), under certain conditions on the matrix \( G \) and the sparsity of \( x \), both (1) and (2) have the same unique solution. The Restricted Isometry Property (RIP) and the coherence of matrix \( G \) are to date the most widely used tools to derive such conditions. A comprehensive introduction to compressive sensing can be found in excellent review papers \([6, 7]\).

Unlike the conventional CS theory which considers only the sparsity of the signal, in \([8]\) a more realistic signal model and CS approach, referred to as the Model-based CS, was introduced by considering some structure among the signal coefficients. Model-based CS and its variations has been widely studied in the literature \([8–14]\). Due to the known structure of the sparse signal in model-based CS, signal models provide two major advantages to CS. First, the number of measurements required for the reconstruction of the sparse signal is significantly reduced. Second, a more accurate and robust recovery is achieved as we can better differentiate true signal information from recovery artifacts \([8]\). In this paper, we are particularly interested in the reconstruction of sparse binary signals using linear measurements. Binary CS is a special case of model based CS which has many applications, such as event detection in wireless sensor networks (WSNs), group testing, and spectrum sensing for cognitive radios. More practical settings and similar discrete types of compressive sensing scenarios can be found in \([15–21]\). More specifically, in binary CS, the nonzero elements of the sparse signal have the same value of 1, and the objective is to minimize the number of real valued measurements linearly generated from the binary sparse signal.

The \( \ell_1 \)-minimization solution for the noiseless binary CS has been proposed in \([17]\), where (2) was solved by limiting the reconstructed signal in the range \([0, 1]\). Also, by using an integer programming, \([15]\) proposed a model to solve a
shown to be of number of required measurements is then approximated and the row weight of the measurement matrix. The minimum optimum measurement degree, defined as the number of nonzero signal elements in the bipartite graph, where $T \geq 0$ is a predefined value. To minimize the number of required measurements. A message passing decoder has been proposed for binary CS problems, where the measurement matrix is sparse and its elements are randomly obtained based on a predefined probability distribution function. However, the number of required measurements in these approaches are still very large, as they do not take the binary structure of the signal into the design of the CS approaches.

In this paper, we propose a binary CS approach based on analog fountain codes (AFCs) [22]. AFCs were originally designed to approach the capacity of the Gaussian channel across a wide range of signal to noise ratios (SNRs). Each AFC coded symbol is a linear combination of randomly selected information symbols with real weight coefficients. With the proper design of the weight coefficients, each binary linear equation associated with each AFC coded symbol will have a unique solution. As shown in [22], the AFC decoding problem is equivalent to solving a system of binary linear equations at high SNRs, which is the same as the reconstruction of a binary sparse signal from noiseless linear measurements. This interesting property of AFC codes motivates us to incorporate them in the design of an efficient binary CS approach to minimize the number of measurements required for the successful recovery of the sparse signal.

In the proposed analog fountain compressive sensing (AFCS) approach, each measurement is generated from a number of randomly selected binary information symbols with weight coefficients selected from a finite weight set. Thanks to the random structure of AFC codes, the number of generated measurements is potentially limitless and the transmitter can keep transmitting the measurements till the destination fully recovers the sparse binary signal. Due to the sparsity and the fact that weight coefficients do not introduce redundancy, each measurement is connected to a very small number of non-zero signal components in the respective weighted bipartite graph; thus, information symbols can be recovered by comparing the measurement value with the weight coefficients assigned to it in an iterative manner. More specifically, we propose a reconstruction algorithm, where each measurement can determine its connected variable nodes if it has connection with at most $T$ nonzero signal elements in the bipartite graph, where $T \geq 0$ is a predefined value. To minimize the number of required measurements, we propose an optimization problem to find the optimum measurement degree, defined as the number of information symbols connected to each measurement or simply the row weight of the measurement matrix. The minimum number of required measurements is then approximated and shown to be of $O(-n \log(1-s))$, where $n$ is the signal length and $s \triangleq k/n$ is the signal sparsity order. Simulation results show that the proposed scheme has a significantly lower error rate compared to the $\ell_1$-minimization approach [17] and the binary CS scheme in [21].

We further extend the proposed AFCS scheme to the noisy CS problem, where the measurements are received in the presence of additive white Gaussian noise (AWGN). We propose a message passing decoder to reconstruct the binary sparse signal and show that the AFCS can recover the sparse signal with a very small number of measurements compared to the $\ell_1$-minimization approach [17]. More specifically, the proposed AFCS scheme requires less number of measurements to completely recover the binary sparse signal in a wide range of SNRs even with a lower measurement degree compared to the $\ell_1$-minimization approach [17].

As a direct application of the AFCS, we then consider sparse event detection (SED) in WSNs, where a set of monitoring sensors try to capture a small number of active events randomly distributed in the sensing field [23]. CS mechanisms are beneficial to WSNs as sensor nodes usually have constrained sensing, limited memory and processing capability, as well as small communication bandwidth [24]. Several approaches have used CS theory for SED problem, but they are impractical in some scenarios. For example, Bayesian compressive sensing (BCS) has been used in [25] to solve the SED problem in WSNs. However, BCS is known to be sensitive to noise, leading to a considerable degraded performance in low SNRs. To solve this problem, [26] applied sequential compressive sensing (SCS) to model the SED problem as a support recovery problem and solve it in an adaptive manner. However, in this approach, a large number of sensor nodes (about half that of event sources) is required to be deployed in the sensing field in order to guarantee its performance. This will also bring the system a significant communication cost due to a large number of sensor nodes.

In this paper, we revisit the sparse event detection problem in WSNs from the coding theory perspective and solve the problem as a decoding problem of AFCS. We formulate the variable and check node degree distributions for two scenarios of random and uniform deployment of sensor nodes. The minimum number of sensor nodes required to fully detect all active events in the noiseless case is then found as a function of the sensing coverage radius and the total area of the sensing field. We show that the proposed AFCS scheme approaches this lower bound in high SNRs. We further analyze the probability of false detection for the proposed scheme and show that the AFCS achieves a very low probability of false detection at different SNRs. Simulation results show that the proposed scheme outperforms the SCS [26] and BCS [25] approaches in terms of the probability of correct detection at different SNRs and the minimum sampling ratio to successfully detect the active event sources at different sampling ratios.

The rest of the paper is organized as follows. In Section II, we introduce the AFCS approach and the general reconstruction algorithm for the AFCS scheme. The analysis of the proposed reconstruction algorithm in the noiseless case is then presented in Section III, followed by the optimization of the measurement degree and an approximation for the number of required measurements. A message passing decoder is then proposed for AFCS in the presence of additive white Gaussian noise in Section IV. In Section V, we demonstrate the application of AFCS in WSNs for sparse events detection. Section VII provides some discussion on the practical issues for sparse event detection in WSNs. Simulation results are shown in Section VI followed by some concluding remarks in Section VII.
## TABLE I
### NOTATION SUMMARY

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>Length of the binary sparse signal (number of event sources)</td>
</tr>
<tr>
<td>$m$</td>
<td>Number of measurements (sensor nodes)</td>
</tr>
<tr>
<td>$k$</td>
<td>Number of nonzero elements of the binary sparse signal (number of active events)</td>
</tr>
<tr>
<td>$s$</td>
<td>Sparsity order, $s = k/n$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Sampling ratio, $\beta = m/n$</td>
</tr>
<tr>
<td>$L$</td>
<td>Measurement (check node) degree</td>
</tr>
<tr>
<td>$d_{r/e}$</td>
<td>Variable node degree</td>
</tr>
<tr>
<td>$b$</td>
<td>Binary sparse event signal of dimension $n$</td>
</tr>
<tr>
<td>$G_{m \times n}$</td>
<td>$m \times n$ measurement matrix, $G = [g_{i,j}]$</td>
</tr>
<tr>
<td>$W$</td>
<td>Weight set of AFCs with size $D$</td>
</tr>
<tr>
<td>$T$</td>
<td>Maximum number of weight coefficients included in each operation of the decoding algorithms</td>
</tr>
<tr>
<td>$S$</td>
<td>Monitoring region of area $S$</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Binary sparse event signal of dimension $n$</td>
</tr>
<tr>
<td>$H_{m \times n}$</td>
<td>$m \times n$ channel matrix, $H = [h_{i,j}]$</td>
</tr>
<tr>
<td>$g_{i,j}$</td>
<td>The channel between the $i^{th}$ sensor node and the $j^{th}$ event source</td>
</tr>
<tr>
<td>$d_{i,j}$</td>
<td>The distance between the $i^{th}$ sensor node and the $j^{th}$ event source</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Propagation loss factor</td>
</tr>
<tr>
<td>$\epsilon_i$</td>
<td>Thermal measurement noise at the $i^{th}$ sensor</td>
</tr>
<tr>
<td>$R$</td>
<td>Sensor coverage radius</td>
</tr>
<tr>
<td>$PFD$</td>
<td>Probability of false detection</td>
</tr>
</tbody>
</table>

## II. ANALOG FOUNTAIN COMPRESSIVE SENSING

### A. Preliminaries

Throughout the paper the following notations are used. We use a boldface small letter to denote a vector and the $i^{th}$ entry of vector $v$ is denoted by $v_i$. A boldface capital letter is used to denote a matrix and the $i^{th}$ element in the $j^{th}$ row of matrix $G$ is denoted by $g_{i,j}$. The transpose of matrix $G$ is denoted by $G'$. A set is denoted by a curly capital letter and the cardinality of set $S$ is represented by $|S|$. We assume that a binary signal vector of length $n$, $b$, is strictly $k$-sparse, which means that the number of non-zero elements of $b$ is exactly $k$. The sparsity order of $b$ is then defined as $s = k/n$. Table I provides a summary of notations for quick reference.

Let us first briefly introduce the analog fountain code (AFC). AFCs were originally proposed in [22, 27] as an effective adaptive transmission scheme to approach the capacity of wireless channels. In AFC, to generate each coded symbol $L$ information symbols are selected uniformly at random and then linearly combined with real weight coefficients. For simplicity, we assume that weight coefficients are chosen from a finite weight set, $W$, based on a predefined probability distribution function, called degree distribution. For simplicity, we assume that the measurement degree, i.e., the number of non-zero elements in each row of the measurement matrix, is a fixed value. Let $g_i$ denote the $i^{th}$ row of the measurement matrix $G$; then, only $L$ entries of $g_i$ have nonzero values. The value of nonzero elements of $g_i$ are also randomly obtained from the weight set $W$ without the repetition. Although the weight coefficients can be obtained based on a predefined probability distribution function, for simplicity we assume that the weight set $W$ has $D \geq L$ positive real members as follows:

$$W = \{w_1, w_2, ..., w_D\}.$$

The measurements are then calculated by $c = Gb$, which can be graphically represented by a weighted bipartite graph with elements of the binary signal $b$ and the measurements as variable and check nodes, respectively. In the bipartite graph, the variable node $j$ is connected to the check node $i$, if $g_{i,j}$ is non-zero, and the edge between them is also assigned with the value of $g_{i,j}$. Fig. 1 shows the weighted bipartite graph of an AFCS with $L = 2$.

As shown in [28], for and AFCs with measurements, the probability that a variable node is not connected to any check node in its bipartite graph is $e^{-mL/n}$, which is due to random selection of variable nodes when generating each measurement. This problem significantly increases the error rate of the CS schemes in [17, 21]. To overcome this problem, we modify the measurement matrix such that each variable node is connected to at least one check node. Moreover, to ensure that all the variable nodes have the same chance to be recovered, the degrees of all variable nodes are chosen to be the same. To achieve this, we select the variable nodes among those with the lowest degrees when generating each measurement. In this way, the variable node degrees will be either $d_v$ or $d_v - 1$, where $d_v$ is the smallest integer larger than or equal to $mL/n$ [28]. Later in Section V, we show how this imposed structure in the measurement matrix can be obtained in real applications, which significantly increases the sensing efficiency and improve the reconstruction capability.

### B. Measurement Matrix Construction

To generate each row of the measurement matrix, first an integer $L$, called measurement degree, is obtained based on a predefined probability mass function, called degree distribution. For simplicity, we assume that the measurement degree, i.e., the number of non-zero elements in each row of the measurement matrix, is a fixed value. Let $g_i$ denote the $i^{th}$ row of the measurement matrix $G$; then, only $L$ entries of $g_i$ have nonzero values. The value of nonzero elements of $g_i$ are also randomly obtained from the weight set $W$ without the repetition. Although the weight coefficients can be obtained based on a predefined probability distribution function, for simplicity we assume that the weight set $W$ has $D \geq L$ positive real members as follows:

$$W = \{w_1, w_2, ..., w_D\}.$$

The measurements are then calculated by $c = Gb$, which can be graphically represented by a weighted bipartite graph with elements of the binary signal $b$ and the measurements as variable and check nodes, respectively. In the bipartite graph, the variable node $j$ is connected to the check node $i$, if $g_{i,j}$ is non-zero, and the edge between them is also assigned with the value of $g_{i,j}$. Fig. 1 shows the weighted bipartite graph of an AFCS with $L = 2$.

As shown in [28], for and AFCs with measurements, the probability that a variable node is not connected to any check node in its bipartite graph is $e^{-mL/n}$, which is due to random selection of variable nodes when generating each measurement. This problem significantly increases the error rate of the CS schemes in [17, 21]. To overcome this problem, we modify the measurement matrix such that each variable node is connected to at least one check node. Moreover, to ensure that all the variable nodes have the same chance to be recovered, the degrees of all variable nodes are chosen to be the same. To achieve this, we select the variable nodes among those with the lowest degrees when generating each measurement. In this way, the variable node degrees will be either $d_v$ or $d_v - 1$, where $d_v$ is the smallest integer larger than or equal to $mL/n$ [28]. Later in Section V, we show how this imposed structure in the measurement matrix can be obtained in real applications, which significantly increases the sensing efficiency and improve the reconstruction capability.

### C. Weight Set Design

Let us assume that for an AFCS of degree 2, $w_1 + w_2 = w_3$ and a measurement has a value of $w_3$ and is connected to $b_1$, $b_2$, and $b_3$ with weight coefficients, $w_1$, $w_2$, and $w_3$, respectively. 

![Weighted bipartite graph of an AFC]
respectively. The decoder then try to solve the binary linear equation \( w_1 b_1 + w_2 b_2 + w_3 b_3 = w_3 \). This measurement can determine either \( b_1 = b_2 = 1, b_3 = 0 \), or \( b_1 = b_2 = 0, b_3 = 1 \), i.e., there are two possible solutions. It is then easy to show that to guarantee that each measurement can uniquely verify its connected variable nodes, it is sufficient that the weight set \( W \) satisfies the following condition:

\[
\sum_{i=1}^{D} v_i w_i \neq 0, \quad (4)
\]

for all \( v \in \{-1, 0, 1\}^D \), where \( v \neq \{0\}^D \), and \( w_i \in W \) for \( i = 1, ..., D \). As stated in [22], each measurement is associated with a binary linear equation and thus the decoder task is actually to solve a system of binary linear equations in the noiseless case. Condition (4) guarantees that the linear equation associated with each measurement has a unique solution.

It is clear that when the members of the weight set are sampled from a continuous probability distribution function, (4) can be easily satisfied. In practice, the elements of the measurement matrix depend on the physical condition of the system, which is mainly random. For example, as we will later see in Section V, in the sparse event detection in WSNs, the elements of the measurement matrix are channel gains, which are random due to shadowing and small scale fading; thus, condition (4) is satisfied and each measurement can uniquely determine its connected variable nodes. In the rest of the paper, we use weight coefficients which have been obtained from a zero mean Gaussian distribution with variance 1, unless otherwise specified.

It is well known that \( \ell_1 \)-minimization reconstructs every sparse signal precisely when the sensing matrix satisfies the null space property (NSP) [9]. In general, the null space of measurement matrix \( G \) is defined as \( \mathcal{N}(G) = \{ z \mid Gz = 0 \} \). If we want to recover all sparse signals \( x \) from the measurements \( Gx \), we must have \( Gx \neq Gx' \) for any pair of distinct vectors \( (x, x') \), since otherwise it would be impossible to distinguish \( x \) from \( x' \). It is worth mentioning that if the weight coefficients in AFCS satisfy condition (4), then each sparse binary signal is uniquely recovered form its linear measurements, which precisely satisfy the NSP (\( \mathcal{N}(G) = \{0\} \)) for all binary signals, which is sufficient for compressive sensing.

### D. Verification Based Reconstruction

It is clear that condition (4) guarantees that a binary linear equation with weight coefficients from \( W \) has a unique solution of zero if and only if the binary variables are all zero [29, 30]. Also due to signal sparsity, only few nonzero signal elements are connected to each measurement. Therefore, by comparing the value of a measurement and the sum of different numbers of its weight coefficients, we can uniquely determine the values of its connected variable nodes. Based on these observations, we then propose the following verification based reconstruction algorithm, referred to as the **Sum Verification Decoder**.

Let us consider the binary compressive sensing problem \( c = Gb \), where the binary sparse signal vector \( b_{n \times 1} \) has to be reconstructed from linear measurements \( c_{m \times 1} \) and a given measurement matrix \( G_{m \times n} \). We use the notation \( \mathcal{N} \) to denote the set of variable nodes connected to check node \( c_i \), and use \( \mathcal{N} \setminus B \) to denote the set of connected variable nodes to check node \( c_i \), excluding the variable nodes in the set \( B \). The operations of the **Sum Verification Decoder** can then be summarized as follows, for a given \( T > 0 \):

**Sum Verification Decoder**: Construct the weighted bipartite graph from the measurement matrix \( G \). Construct a matrix \( G_T \), where its rows correspond to all binary vectors of dimension \( D \) with at most \( T \) nonzero elements. Construct a vector \( w = [w_1, w_2, ..., w_D] \). Compute the vector \( u = G_T w \). Initially, we assume that \( b \) is a zero vector of dimension \( n \).

1. Find a measurement \( c_i \) with a nonzero degree which has a value equal to one of the elements of vector \( u \) (if there is no such measurement, this decoding algorithm halts at this point and fails to recover all the variable nodes).

   Find the set of its connected variable nodes \( \mathcal{N}_i \).

   a) Find the number of elements in \( \mathcal{N}_i \), i.e. \( |\mathcal{N}_i| = \tau \), where \( \mathcal{N}_i = \{j_1, ..., j_\tau\} \). Construct a matrix \( \Gamma \), where its rows correspond to all binary vectors of dimension \( \tau \) with at most \( T \) nonzero elements. Construct a vector \( g_i^{(1)} = [g_{i,1}, g_{i,2}, ..., g_{i,\tau}] \). Compute the vector \( z \) as follows: \( z = \Gamma g_i^{(1)} \). Find \( \ell \) such that \( z_\ell = c_i \).

   b) Set \( b_j = \Gamma_{\ell,r} \) for \( r = 1, 2, ..., \tau \).

2. Remove all the edges connected to verified variable nodes \( b_j \).

3. Update the measurement values to \( c = c - Gb \).

4. Repeat (1), (2), and (3) until all \( b_j \)'s are determined.

In each iteration of the proposed decoder, a measurement which has a connection to at most \( T \) nonzero variable nodes is found (Step 1). Since the weights are chosen from a fixed weight set, this can be achieved by simply comparing the value of the measurements with a table consisting of all the possible values of the summation of at most \( T \) weight coefficients. If such a measurement cannot be found, the decoder will be terminated and the variable nodes cannot be fully recovered. In the next step, the variable nodes that are connected to the selected measurement will be uniquely recovered (Step 1, (a) and (b)). Then, the edges connected to the recovered variable nodes are removed from the bipartite graph (Step 2). Also, the measurement values will be updated to cancel out the effect of the recovered variable nodes on the remaining measurements (Step 3). This algorithm is repeated until all the variable nodes are recovered.

Since the signal is sparse, a majority of measurements are connected to a very small number of nonzero variable nodes. Therefore, it is reasonable to only consider small values of \( T \), which also avoids huge complexity at the decoder. More specifically, in this paper we only consider \( T = 0, 1, \), and 2. Fig. 2 shows the proposed reconstruction algorithm for the case that the measurement degree is \( L = 4 \) and \( T = 2 \). As can be seen in this figure, measurement \( c_i = g_{i,1} + g_{i,3} \) is connected to \( b_1, b_2, b_3, \) and \( b_{n-2} \), and by comparing the measurement value with vector \( z \), all connected variable nodes can be uniquely determined.
III. PERFORMANCE ANALYSIS OF THE VERIFICATION BASED DECODER

In this section, the performance of the sum verification decoder is analyzed for different values of $T$. We then find the optimum measurement degree as a function of the sparsity order, $s$. An upper bound is also derived for the number of measurements required to successfully recover all variable nodes.

A. Analysis of the Verification Probability

Let $G$ denote the bipartite graph corresponding to the code at the receiver. We choose subgraph $G_{\ell}$ as following. Choose an edge $(v, w)$ uniformly at random from all edges. Call the variable node $v$ the root of $G_{\ell}$. Subgraph $G_{\ell}$ is the graph induced by $v$ and all neighbors of $v$ within distance $2\ell$ after removing the edge $(v, w)$. As shown in [31], $G_{\ell}$ is a tree asymptotically, so we refer to $G_{\ell}$ as a SUM-OR tree, where SUM and OR nodes refer to check and variable nodes, respectively. Fig. 3 shows the SUM-OR tree of depth $2\ell$, where the root of the tree is at depth 0 and has children at depth 1. More specifically, a node at depth $t$ has children at depth $t+1$ for $t = 0, ..., 2\ell-1$ and nodes at depths 0, 2, ..., $2\ell$ are referred to as OR nodes, while nodes at depths 1, 3, ..., $2\ell-1$ are referred to as SUM nodes.

In the following, we rephrase the proposed sum verification decoder by using the SUM-OR tree. At every iteration of the algorithm, messages (0 or 1) are passed along the edges from check nodes to variable nodes and vice versa. A variable node sends 1 to a neighboring check node if and only if it has received at least one message with value 1 from its other neighboring check nodes. Also, a check node sends 0 to a neighboring variable node if and only if it is connected to more than $T$ unrecovered variable nodes.

Since, a subgraph expanded from each variable node is asymptotically a tree, then the message exchange between variable and check nodes in the bipartite graph can be seen as the message exchange between SUM and OR nodes in the tree. Therefore, if we map the first iteration of the decoding algorithm to the message exchange between OR and SUM nodes at depths $2\ell$ and $2\ell-1$, respectively, then after $\ell$ iterations, the root of the tree will receive messages from its children. The decoding error probability is then the probability that the root of the tree does not receive at least one message of value 1 from its children. The following lemma gives the decoding probability which is equivalent to the probability that the root of the SUM-OR tree does not receive a message of value 1 from its children.

**Lemma 1:** We consider the binary AFCS, where the measurements are generated by the AFCS process. Let $p_1$ denote the probability that a variable node is recovered after $\ell$ iterations of the sum verification decoder, where $p_0 = 0$. Let $q_0^{(0)}$ and $q_0^{(1)}$ denote the probability that an unrecovered variable node in the $\ell$th iteration of the reconstruction algorithm is zero and one, respectively, where $q_0^{(1)} = 1 - q_0^{(0)}$. Then for a fixed measurement degree $L$, $p_1$ is calculated as follows:

$$p_1 = 1 - \delta(1 - f_1),$$

where $\delta(x) = \sum_j \delta_j x^j$.

$$f_1 = \frac{d}{d-i} \left( \begin{array}{c} d-i \cr j \end{array} \right) p_{i-1} (q_{i-1}^{(0)} (1 - p_{i-1}))^{d-i} \lambda_{i-1}^j$$

$$+ \frac{d-T}{d-i} \left( \begin{array}{c} d-i \cr T \end{array} \right) q_{i-1}^{(0)} p_{i-1} (q_{i-1}^{(0)} (1 - p_{i-1}))^{d-i} \lambda_{i-1}^j,$$

$$d = L - 1, \lambda_i = q_i^{(1)}/q_i^{(0)}, A = \min\{T-1, d-i\}, q_i^{(0)} = 1 - \frac{A}{T},$$

and $\delta(x)$ is the variable node degree distribution function with respect to the edges.

**Proof:** See Appendix A in [32].

To complete the analysis, we only need to find $\delta(x)$. As we assume that each measurement is of degree $L$, variable nodes are either of degree $d_v$ or $d_v - 1$, where $d_v$ is the smallest integer larger than or equal to $\beta L$ and $\beta$ is the sampling ratio, defined as the ratio of the number of measurements and that of variable nodes. Let $v_1$ and $v_2$ denote the probability that a variable node is of degree $d_v$ and $d_v - 1$, respectively, then $v_1 + v_2 = 1$ and

$$v_1 d_v + v_2 (d_v - 1) = \beta L,$$

which results in $v_1 = 1 - d_v + \beta L$ and $v_2 = d_v - \beta L$. Thus, the variable node degree distribution, $\Delta(x)$, can be written as follows:

$$\Delta(x) = (1 - d_v + \beta L) x^{d_v} + (d_v - \beta L) x^{d_v - 1}.$$  

Accordingly, $\delta(x)$, can be calculated as follows [31]:

$$\delta(x) = \frac{\Delta'(x)}{\Delta(1)} = \frac{v_1 d_v x^{d_v - 1} + v_2 (d_v - 1) x^{d_v - 2}}{\beta L}.$$  

1All proofs of this paper are omitted due to page limit and interested readers can refer to our full version [32] for details.
where $\Delta'(x)$ is the derivative of $\Delta(x)$ with respect to $x$.

It is important to note that if $T = L$, then according to (6) we have $f_1 = 1$, and accordingly $p_4 = 1$. This means that each check node can uniquely determine its connected variable nodes if the parameter $T$ equals to the measurement degree in the sum verification decoder.

B. Optimization of the Measurement Degree

According to Lemma 1, we can formulate an optimization problem to find the optimum measurement degree in order to minimize the number of measurements required for the successful recovery of all variable nodes. However, as can be seen in this lemma, the verification probability is not a linear function of the measurement degree; thus, finding the optimum degree is not straightforward. To overcome this problem, we propose a suboptimum solution to find the measurement degree to maximize the number of recovered variable nodes in each iteration of the sum verification decoder. More specifically, in the first iteration of the decoding algorithm, we have $p_0 = 0$, $q_0^{(0)} = 1 - k/n$, and $\lambda_0 = \frac{k}{n} / (1 - \frac{k}{n})$; thus, we have:

$$f_1 = \left( d \right)^T \left( q_0^{(0)} \right)^L \lambda_0^T + \sum_{j=0}^{T-1} \left( L - 1 \right)^j \left( q_0^{(0)} \right)^{L-1} \lambda_0^j.$$  \hspace{1cm} (8)

In other words, a measurement can determine its connected variable nodes if the number of nonzero variable nodes connected to that measurement is at most $T$. Thus, (8) can be rewritten as follows:

$$f_1 = \sum_{j=0}^{T} \left( \frac{L}{j} \right)^j \left( q_0^{(0)} \right)^{L-j} \left( q_0^{(1)} \right)^j.$$  \hspace{1cm} (9)

Let $q_0 \triangleq q_0^{(0)}$ which equals to $1 - k/n$, then the average number of variable nodes, $R$, which can be verified by each measurement, can be calculated as follows:

$$R = \sum_{j=0}^{T} \left( \frac{L}{j} \right)^j q_0^{L-j} (1 - q_0)^j.$$  \hspace{1cm} (10)

The following lemma gives the optimum measurement degree, $L_{opt}$, for a given $T$ and $s$.

**Lemma 2**: The optimal measurement degree, $L_{opt}$, to maximize the average number of variable nodes, which can be verified by each measurement for a given $T$ and sparsity order $s$ in the sum verification decoder, can be approximated as follows:

$$L_{opt} \approx \left\lfloor -\frac{T + 2}{2 \log(1 - s)} \right\rfloor.$$  \hspace{1cm} (11)

where $\lfloor x \rfloor$ is the ceil operator.

**Proof**: See Appendix B in [32].

From (11) it can be noted that the optimum measurement degree is a decreasing function of the sparsity order, $s$, as $|\log(1 - s)|$ is an increasing function of $s$. This means that to guarantee that a measurement can verify the maximum number of variable nodes form a very sparse signal, the measurement degree should be very large. Moreover, a binary signal with a higher $s$ require a lower measurement degree, as the number of nonzero variable nodes is relatively high;

C. The number of measurements

In the proposed reconstruction algorithm, we can verify a large number of variable nodes, including zero and nonzero signal components, from each measurement because a majority of the measurements are connected to a very small number of non-zero elements of the sparse signal. The following lemma gives an upper bound on the number of measurements required to recover the sparse signal by using the proposed reconstruction algorithm.

**Lemma 3**: We consider the binary AFCS, where the measurements are generated by the AFCS process with the measurement degree $L_{opt}$ calculated in (11). Let $m$ denote the minimum number of measurements required for the signal of length $n$ and sparsity order $s$, to be completely reconstructed by the sum verification decoder. Then, $m$ is of $O( - n \log(1 - s) )$.

**Proof**: See Appendix C in [32].

It can be easily proven that the propose scheme requires fewer measurements compared to the existing binary CS approach [21], which the number of required measurements is of $O(k \log(n))$. This is because $\log(x)/x$ is a decreasing function of $x$ when $x \geq e$; thus, $-n \log(1 - s) \leq k \log(n)$ for $k \leq n - e$.

D. Performance Evaluation of noiseless AFCS

![Fig. 4. The optimum measurement degree versus the sparsity order of the signal. Solid lines show the optimum values, and dashed line shows the approximation which has been obtained from (11).](Image)

From Fig. 5 it can be seen that the AFCS scheme with $L = 20$
achieves the error rate of $10^{-4}$; however, the error rate of the binary CS approach in [21] with the measurement degree $L = 30$ is in the order of $10^{-1}$, and the error rate in binary $\ell_1$-minimization [17] is in the order of $10^{-2}$. In other words, the AFC scheme with an even smaller measurement degree outperforms the existing binary CS schemes. The reason is that in $\ell_1$-minimization approach and binary CS scheme in [21], the variable nodes are selected uniformly at random to generate each measurement; thus, even with a large number of measurements some variable nodes may still not be selected by any measurement, leading to a higher error rate. However, in AFC, the variable node degrees are uniform so the probability that a variable node is not selected by any check node is zero. Thus, AFC scheme can approach very small error probabilities even with a smaller measurement degree and decoding parameter $T$. Furthermore, as can be seen in Fig. 5, the proposed AFC scheme with $L = 25$ has a lower error rate compared to the AFC scheme with $L = 20$ and $L = 30$, due to the fact that the optimum measurement degree for sparsity order $s = 0.1$ and $T = 1$ is about $L = 25$; thus, with measurement degrees higher or lower than $L = 25$, the error performance will be degraded.

Fig. 6 shows the required sampling ratio to successfully recover the binary sparse signal versus the sparsity order. As can be seen in Fig. 6, the number of measurements for the proposed AFCS scheme is significantly lower than that for the binary CS scheme in [21]. In fact, the number of measurements for the binary CS scheme in [21] is of $O(k \log(n))$ and that for the proposed AFCS scheme is of $O(-n \log(1 - \frac{1}{n}))$.

It is important to note that in the proposed reconstruction algorithm, the maximum value of $T$ is considered to be 2. Thus, for each measurement the total number of $1 + L + \binom{L}{2}$ comparisons should be performed to recover the connected variable nodes. As a result, the total number of comparisons is of $O(L^2)$. As we have shown that the number of required measurements is of $O(-n \log(1 - s))$ and the optimum measurement degree is $L_{opt} \approx -1/\log(1 - s)$, then the reconstruction complexity of the proposed algorithm is of $O(-n/\log(1 - s))$. This indicates that the complexity of the proposed algorithm increases when the number of nonzero elements of the sparse signal is very small. In fact when $s$ goes to zero, $-1/\log(1 - s)$ can be approximated by $1/s$, so the complexity of the decoder will be of $O(n^2/k)$.

IV. NOISY BINARY COMPRESSIVE SENSING

In this section, we consider the problem of binary compressive sensing in the presence of additive white Gaussian noise (AWGN). In the presence of the noise, AFCS measurements are given by

$$ y = Gb + z. $$

where $z$ is an $m$ by 1 vector of zero mean additive white Gaussian noise with variance $\sigma_z^2 I_m$ and $I_m$ is the $m \times m$ identity matrix. In [22], we proposed a belief propagation (BP) decoder for analog fountain codes, which is actually a modification of the decoding algorithm originally proposed for sparse signal reconstruction in [33]. More details on this decoder can be found in [32].

Fig. 7 shows the minimum sampling ratio ($\beta$) required for the successful reconstruction of all $k$-sparse binary signals ($k = 100$) of length $n = 1000$ versus the SNR. As can be seen in this figure, by increasing the measurement degree $L$ the performance of the AFCS scheme is improved. More specifically, when $SNR = 30$ dB, the AFCS scheme with $L = 12$ requires 160 measurements, while it requires 200 and 240 measurements when $L = 10$ and $L = 8$, respectively. Note that the reconstruction algorithm proposed in [21] only works in the noiseless settings. So, we only compare the proposed AFCS scheme with the $\ell_1$-minimization approach [17] for the noisy case. As can be seen in Fig. 7, the AFCS scheme with even a lower measurement degree always outperforms the $\ell_1$-minimization approach across all SNR values. The minimum sampling ratio for the proposed AFCS approach with non-uniform measurement matrix, where the variable node degree is not fixed, is also shown in Fig. 7. As can be seen in this figure, the proposed AFCS approach with non-uniform variable node degrees and a very smaller measurement degree ($L = 12$) performs very close to the binary $\ell_1$ scheme with a very large degree ($L = 30$). This shows the superiority of the proposed scheme over conventional binary CS scheme for different measurement matrices and SNR values.
\[ y_j = \sum_{i \in S_j} h_{i,j} e_i + \epsilon_j, \quad (13) \]

where \( S_j \) is the set of event sources inside the sensing range of the \( j \)th sensor node, and \( \epsilon_j \) is the thermal measurement noise at the \( j \)th sensor node. The received signal vector \( y_{m \times 1} \) can then be written as follows:

\[ y_{m \times 1} = H_{m \times n} e_{n \times 1} + \epsilon_{m \times 1}, \quad (14) \]

where \( H \triangleq [h_{i,j}]_{i,j} \) and \( e = [e_i]_{1 \leq i \leq n} \).

The sink node is assumed to have full knowledge of the channel matrix \( H \) and all sensors’ readings are sent to the sink node via a wireless channel at SNR \( \gamma \) [16]. More specifically, we assume the whole network is fully connected, which means that the sensor nodes can communicate with each other in the local communication range and each sensor node can correctly receive its neighbor’s message [16]. Generally, we assume that sensors’ readings are relayed by the sensor nodes towards the sink node and they will be received at the sink node at SNR \( \gamma \). The received signal at the sink node can then be shown as follows:

\[ x_{m \times 1} = y_{m \times 1} + z_{m \times 1}, \quad (15) \]

where \( z \) is the \( m \times 1 \) additive white Gaussian noise (AWGN) vector with zero mean and variance \( \sigma_z^2 \), and \( \gamma = ||He||^2/\sigma_z^2 \). This model has been widely used for sparse event detection in WSNs [25, 26, 34]. We will further discuss about the practical issues of this model in Section VI and propose some approach for its practical implementation.

**B. Sparse Event Detection in WSNs as an AFCS**

Since event sources generate binary signals, the event vector \( e \) can be considered as a vector of binary information symbols. As we assume that each sensor node receives signals only from active event sources inside the sensor’s coverage area, the number of nonzero entries in each row of \( H \) is small. Thus the sensing process in (15) is equivalent to the measurement process in AFCS, where the channel gains are considered as weight coefficients in the measurement matrix. More specifically, the sensor degree is defined as the number of event sources inside the coverage area of a sensor node, determining the number of nonzero elements in the respective row of the measurement matrix. We further define the event degree as the number of sensor nodes which have covered that event source. The following lemma gives the probability distribution of the sensor degree in the uniform deployment scenario and gives a lower bound on the minimum number of sensor nodes required for having a minimum event degree of at least 1.

**Lemma 4:** Let \( n \) denote the number of event sources in the monitoring region \( S \), and let \( m \) denote the number of sensor nodes with coverage radius \( R_s \) uniformly placed inside \( S \) to detect active event sources. Let \( \Omega_{u,d} \) denote the probability that the sensor degree is \( d \). Then, \( \Omega_{u,d} \) is given by:

\[ \Omega_{u,d} = \alpha_1 f(n,d,P) + \alpha_2 f(n,d,P/P_4) + \alpha_3 f(n,d,P/P_2), \quad (16) \]

where \( P = \pi R_s^2 / S, f(n,d,P) \triangleq \binom{n}{d} P^d (1-P)^{n-d}, \alpha_1 = (m - 4 + 4(\sqrt{m} - 2))/m, \alpha_2 = 4/m, \) and \( \alpha_3 = (m - 4(\sqrt{m} - 2))/m \). Moreover, to guarantee that each event source is covered by at least one sensor node in the uniform deployment scenario, the following condition should be satisfied:

\[ m \geq \left( \frac{\sqrt{\gamma}}{2R_s} \right)^2 + \left( \frac{\sqrt{\gamma}}{2R_s} + 1 \right)^2, \quad (17) \]

where \( \lceil \cdot \rceil \) and \( \lfloor \cdot \rfloor \) are the ceiling and floor operators, respectively.
Proof: See Appendix D in [32].

In the random deployment scenario, where both event sources and sensor nodes are randomly placed inside the sensing field, the sensor degree distribution \( \Omega_{r,d} \) and event degree distribution \( \Lambda_{r,d} \), defined as the probability mass function of the sensor degree and that of event degree, respectively, are given by:

\[
\Omega_{r,d} = \left( \frac{m}{d} \right) P^d (1 - P)^{n-d}, \quad \Lambda_{r,d} = \left( \frac{n}{d} \right) P^d (1 - P)^{m-d},
\]

where \( P = \pi R_e^2 / S \). This arises from the fact that the probability that an event source is located inside the coverage area of a sensor node is the ratio of the coverage area of the sensor and area of the monitoring region. Since the locations of sensor nodes and event sources are independent, the number of event sources inside the coverage area of a sensor node is a binomial random variable. Similarly, we can show that the event degree follows a binomial distribution with parameter \( m \) and success probability \( P \).

It is important to note that in the random deployment scenario, the probability that an event source is not covered by any sensor node is \( \Lambda_{r,0} = (1 - P)^m \). It is then straightforward to show that in the random deployment scenario, to guarantee that each event source is covered by at least one sensor node with the probability of at least \( 1 - \zeta \), i.e., \( \Lambda_0 \leq \zeta \), the number of sensor nodes should satisfy the following condition:

\[
m \geq \frac{\ln(\zeta)}{\ln(1 - P)},
\]

where \( \zeta > 0 \) is a given real number.

C. Analysis of the False Detection Probability

As previously stated, the objective of the decoding algorithm is to find the conditional probabilities of the variable nodes for a given measurement matrix and the received signals. Since we consider the additive white Gaussian noise with variance \( \sigma_z^2 \), the conditional probabilities can be written as follows:

\[
p(\mathbf{e} | \mathbf{H}) = p(\mathbf{z} = \mathbf{x} - \mathbf{He} | \mathbf{H}, \mathbf{x}) = \left( \frac{1}{\sqrt{2\pi\sigma_z^2}} \right)^m \prod_{j=1}^m \exp \left( -\frac{(x_j - \sum_{i=1}^n h_{j,i} e_i)^2}{2\sigma_z^2} \right),
\]

and the decoding objective is to find \( \mathbf{e} \) in a way that \( p(\mathbf{e} | \mathbf{H}) \) or equivalently \( \log(p(\mathbf{e} | \mathbf{H})) \) is maximized. Then, we have [22]:

\[
\hat{\mathbf{e}} = \arg \min_{\mathbf{e}} ||\mathbf{x} - \mathbf{He}||_2^2.
\]

The following lemma gives an upper bound on the probability of false detection (PFD), defined as the probability that a non-active event is falsely detected as active.

**Lemma 5:** Let \( p_e \) denote the probability of false detection in a WSN consisting of \( m \) sensor nodes with coverage radius \( R_s \), which are placed in a monitoring region of area \( S \). Then, \( p_e \) is upper bounded as follows for a given noise variance \( \sigma_z^2 \):

\[
p_e \leq Q \left( \frac{1}{2\sigma_z} \sqrt{mP(1 - P)R_s^2} \right),
\]

where \( P = \pi R_e^2 / S \) and \( Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} \, du \).

Proof: See Appendix E in [32].

It is important to note that from (20) we can conclude that the probability of false detection in our proposed approach is very small. As an example, for a WSN consisting of \( m = 50 \) sensors randomly placed in \( 500 \, m \) by \( 500 \, m \) monitoring area with coverage radius \( R_s = 50 \, m \), the probability of false detection is about \( 10^{-5} \) at \( \gamma = 20 \, \text{dB} \).

VI. PRACTICAL CONSIDERATIONS IN THE SPARSE EVENT DETECTION IN WSNs

Applying compressive sensing techniques has been utilized by only several existing schemes to event detection in sensor networks [35, 36]. However, all of them directly apply the conventional compressive sensing algorithms. For compressive sensing to be applicable, they assume that the possible locations at which events may occur are known and the number of active events is also available. Furthermore, they also make the impractical assumption that each node is able to monitor every possible location. This essentially means that any sensor node covers the whole monitoring area, which is not true in reality. Therefore, these existing schemes are inapplicable for detection and localization of real-world events that may occur anywhere in the monitoring area, and cover limited region due to quick signal attenuation. More importantly, the number of occurred events is changing over time and unknown in advance.

To relax all these assumptions on the system model, we assume that the monitoring region has an area of \( S \) and is virtually divided into \( n = n_x \times n_y \) small grids as shown in Fig. 8. Thus, the location of an event can be approximated by the centroid of the grid containing the event. Let \( S_j \) denote the set of grids inside the coverage area of sensor node \( j \), then the received signal at sensor node \( j \) can be shown as follows:

\[
y_j = \eta \sum_{i \in S_j} \frac{e_i}{d_{i,j}^{\alpha/2}} + \epsilon_j,
\]

where \( \epsilon_j \) is the thermal measurement noise at the \( j^{th} \) sensor node, \( d_{i,j} \) is the distance between the sensor node \( j \) and the center of grid \( i \), \( \alpha \) is the propagation loss factor, \( \eta \) is a fixed signal gain factor of sensor nodes, and \( e_i = 1 \) if an active event is located in grid \( i \), otherwise it is zero. This model has been widely used in WSNs [37–40] and verified through practical experiments in [36] for the environment temperature. This model can be further used when we assume that event sources are randomly distributed in the sensing field and event sources generate a signal of the same power \( P_0 \) when they are active. The measured signal power at distance \( d \) of each event source can also be modeled by \( P_d = P_0 / d^\alpha \), where \( \alpha \) is the path loss exponent. This model can be used in the environment that the wireless channel is modeled by path loss.

Sensor nodes transmit their observations along with their IDs to the sink node in a time divided manner, in which the total transmission time is divided into \( m \) time slots and sensor \( j \) transmits its observation in the \( j^{th} \) time slot. The channel between sensor node \( j \) and the sink node is known at the sink node which can be simply obtained through the periodical
channel estimation at the sink node. Since the channel varies very slowly in sensor networks, the accurate channel state information can be acquired by pilot symbols in WSN [41]. It is important to note that as the sink node knows the location of each sensor node, it knows the grids which are inside the coverage area of each sensor node. Then, it can calculate the distance between the sensor nodes and the centre of each grid. We can also assume that each sensor node knows its location through embedded Global Positioning System (GPS) receivers or some localization algorithms [36]. Many other multi-hop transmission schemes can also be considered to deliver the measurements to the sink node, which are commonly used in WSNs. As these approaches have been widely studied in the literature and are out of scope of this paper, we refer the interested readers to [36] and references therein for further details.

We can then use the same system model as in (14), where \( h_{i,j} = \eta / d^{\alpha/2}_{i,j} \) and \( n \) is the number of grids. In fact, the sparse event detection can be modeled as a binary compressive sensing, where the number of measurements equals to the number of sensor nodes and the length of the binary sparse signal equals to the number of grids. It is then obvious that by increasing the number of grids, each active event falls inside one and only one grid. Since the number of active events is fixed (say \( k \)), by increasing the number of grids \( n \), the binary vector \( \mathbf{e} \) becomes sparse.

We also make the practical assumption that the sensor network is roughly time synchronized. Note that we do not require accurate time synchronization. Many existing algorithms have been developed for practical time synchronization in large scale sensor networks. For example, FTSP [36] can achieve clock synchronization accuracy as much as 2.24 \( \mu \)s by exchanging a few bytes among neighbors every 15 minutes. Since the event detection period is normally tens of seconds to tens of minutes. Thus, FTSP is sufficient for providing the needed time synchronization accuracy [36].

It is very important to note that we assume that event sources are randomly distributed in the sensing field and only a small number of them are active. We also assume that each event source is active independently of other event sources. However, the measured signals at sensor nodes are correlated, as each sensor node has a sensing range which covers a small area of the sensing field which can overlap with other sensors’ coverage areas. As we assume that the signal of interest is binary, i.e., each event source generates a signal with the known power when it is active, this correlation can be captured in the respective bipartite graph at the sink node. If we assume that sensor nodes and the centroid of grids are shown respectively by check and variable nodes, then a variable node is connected to a check node if and only if the respective grid is located in the coverage area of the respective sensor node. In this way, sensor nodes which are closer to each other have more overlapped area, thus they have more common variable nodes in the bipartite graph.

VII. SIMULATION RESULTS

In this section, we investigate the performance of the proposed AFCS scheme for sparse event detection in WSNs. More specifically, we consider two cases. First, a noiseless case is considered, where the sensor readings are noiseless, so the proposed sum verification decoder can be effectively used for the signal reconstruction from noiseless measurements. Second, we consider a general noisy case and simulate the performance of the AFCS scheme for different SNRs and signal sparsity orders, where the belief propagation decoder is used for the signal reconstruction. Here, the sampling ratio, \( \beta \), is defined as the ratio of the number of sensors, \( m \), and that of event sources, \( n \). We use the probability of correct detection (PCD) as the performance metric, where PCD is defined as the percentage of correctly detected active events [26].

A. SED with Noiseless Measurements

Fig. 9 shows the probability of correct detection as a function of signal sparsity order and sampling ratio, when \( R_s = 50 \) m and \( n = 256 \) event sources are randomly placed inside a 500 m by 500 m sensing field.
sensors' readings to detect all active events. The reason behind this is that when the number of active events increases, the number of active events connected to each sensor node also increases, thus the sum verification decoder with small \( T \) cannot detect them. Moreover, when the sensors are randomly distributed, the degree of each event source follows the Poisson distribution, which results in a non-uniform error protection of the event sources. In other words, a fraction of event sources may be covered by a very small number of sensors; thus, they have a lower chance to be recovered from the measurements. It is important to note that the probability of false detection (PFD) is zero for the proposed sum verification decoder as it only detects the active events and other events are always detected as zero.

Fig. 10 shows the PCD versus \( R_s \) and the sampling ratio for the case that \( n = 256 \) events sources are randomly distributed within a 500 m by 500 m sensing field, where only \( k = 10 \) event sources are active. As can be seen in this figure, the proposed sum verification decoder can completely detect the active events in the uniform deployment scenario even at very small sampling ratios when the coverage radius of sensors is larger than 40m. In fact, by increasing the sensing coverage radius, active event sources are covered by more sensor nodes and thus they can be detected with a higher probability. However, as can be seen in this figure, in the random deployment scenario active event sources cannot be completely detected even with a large sensing coverage radius or high sampling ratio.

Fig. 11 shows the minimum sampling ratio required to successfully detect all active events versus the signal sparsity order \( s \) and \( R_s \) for the uniform deployment scenario. As can be seen in this figure, with increasing \( s \), a larger number of sensors have to be deployed in the sensing field. This is because of the fact that a larger number of active event sources are connected to each sensor nodes; thus, the proposed sum verification decoder with \( T = 2 \) requires a larger number of measurements for sparse event detection. Moreover, when \( R_s \) increases, a smaller number of sensor nodes is required to be deployed in the sensing field to successfully recover all active events. However, when the signal sparsity order increases, a lower number of sensor nodes is required when the sensor coverage radius is low compared to the case that the sensing coverage radius is high.

B. SED with Noisy Measurements

We now compare the performance of AFCS with Bayesian compressive sensing (BCS) [25] and sequential compressive sensing (SCS) [26] at different SNRs and sampling ratios. Fig. 12 shows PCD versus the sampling ratio at different SNR values, when \( n = 1000 \) event sources are randomly distributed in an area of 1000 m by 1000 m, \( k = 10 \), and \( \alpha = 3 \). As can be seen in this figure, the proposed AFCS scheme outperforms both BCS [25] and SCS [26] in terms of PCD in different SNRs. Moreover, the number of required sensor nodes in SCS is relatively large (500), which significantly increases the total number of transmissions, leading to poor energy efficiency. It is important to note that the existing approaches need to know the number of active events to achieve a relatively good performance in terms of PCD; however, the proposed AFCS approach does not need to know the sparsity level of the events and can work for various numbers of active events.

Fig. 13 shows the minimum sampling ratio versus SNR in order to achieve a PCD larger than or equal to 99%, when \( \alpha = 3 \) and \( n = 256 \) event sources are randomly distributed within 500 m by 500 m area. As can be seen in this figure, in the uniform and random deployment scenarios, the minimum number of nodes approaches the lower bounds in (17) and (18), when SNR increases. Moreover, in low SNRs, the proposed approach can detect the active events with a large number of sensor nodes, where a WSN with sensors with a high coverage radius is more robust to the noise effect in terms of the sampling ratio.

VIII. Conclusion

In this paper, we proposed an analog fountain compressive sensing (AFCS) approach for sparse recovery of the
binary sparse signal. We proposed a generalized verification based reconstruction algorithm, which enables a measurement connected to at most \( T \) nonzero signal elements to recover their connected variable nodes. We showed that the number of measurements required for the successful recovery of the sparse signal is of \( O(\sqrt{n} \log(1-s)) \), where \( n \) is the signal length and \( s \) is the sparsity order. Simulation results showed that the AFCS scheme significantly outperforms the conventional binary CS and \( \ell_1 \)-minimization approaches in terms of the error rate for different sparsity orders. We further showed that the sparse event detection problem in wireless sensor networks can be represented by AFCS. Simulation results showed that the proposed AFCS approach outperforms the existing sparse event detection schemes in WSNs in terms of the probability of correct detection in a wide range of signal to noise ratios with a much smaller number of sensor nodes and a negligible probability of false detection.

**REFERENCES**


[27] M. Shirvani Moghadam, Y. Li, and B. Vucetic, “Adaptive analog foun-


Mahyar Shirvanimoghaddam received the B. Sc. degree with 1st Class Honours from University of Tehran, Iran, in September 2008, the M. Sc. Degree with 1st Class Honours from Sharif University of Technology, Iran, in October 2010, and the Ph.D. degree from The University of Sydney, Australia, in January 2015, all in Electrical Engineering. He was a recipient of University of Sydney International Scholarship (USydis), University of Sydney Postgraduate Award (UPA), and University of Sydney Norman I prize. He is currently a Postdoctoral Research Fellow at the School of Electrical Engineering and Computer Science, at The University of Newcastle, Australia.

His general research interests include channel coding techniques, cooperative communications, compressed sensing, machine-to-machine communications, and wireless sensor networks. Mahyar Shirvanimoghaddam is currently serving as the Reviewer of several international prestigious journals, e.g., IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEE Communication Letters, IEEE Transactions on Vehicular Technology, and several international flagship conferences.

Yonghui Li (M04-SM09) received his Ph.D degree in November 2002 from Beijing University of Aeronautics and Astronautics. From 1999 2003, he was affiliated with Linkair Communication Inc, where he held a position of project manager with responsibility for the design of physical layer solutions for the LAS-CDMA system. Since 2003, he has been with the Centre of Excellence in Telecommunications, the University of Sydney, Australia. He is now an Associate Professor in School of Electrical and Information Engineering, University of Sydney. He is the recipient of the Australian Queen Elizabeth II Fellowship in 2008 and the Australian Future Fellowship in 2012.

His current research interests are in the area of wireless communications, with a particular focus on MIMO, cooperative communications, coding techniques and wireless sensor networks. He holds a number of patents granted and pending in these fields. He is an executive editor for European Transactions on Telecommunications (ETT). He received the best paper awards from IEEE International Conference on Communications (ICC) 2014 and IEEE Wireless Days Conferences (WD) 2014.

Jinhong Yuan (M’02–SM’11) received the B.E. and Ph.D. degrees in electronics engineering from the Beijing Institute of Technology, Beijing, China, in 1991 and 1997, respectively. From 1997 to 1999, he was a Research Fellow with the School of Electrical Engineering, University of Sydney, Sydney, Australia. In 2000, he joined the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, Australia, where he is currently a Telecommunications Professor with the School. He has published two books, three book chapters, over 200 papers in telecommunications journals and conference proceedings, and 40 industrial reports. He is a co-inventor of one patent on MIMO systems and two patents on low-density-parity-check codes. He has co-authored three Best Paper Awards and one Best Poster Award, including the Best Paper Award from the IEEE Wireless Communications and Networking Conference, Cancun, Mexico, in 2011, and the Best Paper Award from the IEEE International Symposium on Wireless Communications Systems, Trondheim, Norway, in 2007.

He is currently serving as an Associate Editor for the IEEE Transactions on Communications. He served as the IEEE NSW Chair of Joint Communications/Signal Processings/Ocean Engineering Chapter during 2011-2014. His current research interests include error control coding and information theory, communication theory, and wireless communications.