

Article

An Efficient Compressive Sensing Event-Detection Scheme for Internet of Things System Based on Sparse-Graph Codes

Jun Cai , Xin Xu, Hongpeng Zhu and Jian Cheng *

College of Communications Engineering, Army Engineering University of PLA, Nanjing 210007, China

* Correspondence: chjatrice@163.com

Abstract: This work studied the event-detection problem in an Internet of Things (IoT) system, where a group of sensor nodes are placed in the region of interest to capture sparse active event sources. Using compressive sensing (CS), the event-detection problem is modeled as recovering the high-dimensional integer-valued sparse signal from incomplete linear measurements. We show that the sensing process in IoT system produces an equivalent integer CS using sparse graph codes at the sink node, for which one can devise a simple deterministic construction of a sparse measurement matrix and an efficient integer-valued signal recovery algorithm. We validated the determined measurement matrix, uniquely determined the signal coefficients, and performed an asymptotic analysis to examine the performance of the proposed approach, namely event detection with integer sum peeling (ISP), with the density evolution method. Simulation results show that the proposed ISP approach achieves a significantly higher performance compared to existing literature at various simulation scenario and match that of the theoretical results.

Keywords: event detection; integer-valued signal; IoT; sparse graph codes; compressive sensing



Citation: Cai, J.; Xu, X.; Zhu, H.; Cheng, J. An Efficient Compressive Sensing Event-Detection Scheme for Internet of Things System Based on Sparse-Graph Codes. *Sensors* **2023**, *23*, 4620. <https://doi.org/10.3390/s23104620>

Academic Editors: Wei Yi, Luigi Ferrigno and Nikos Fotiou

Received: 11 October 2022

Revised: 8 February 2023

Accepted: 22 February 2023

Published: 10 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recently, compressive sensing (CS) [1,2] theory has been widely used in the Internet of Things (IoT) system [3–6] and has shown significant performance improvements in network life, energy efficiency, and overall system throughput. CS theory is profitable to IoT system since it usually has limited perception, limited memory, and a small communication bandwidth and capacity [7–9].

In this paper, we investigate how to employ CS in wireless sensor networks, which involve a large number of sensor nodes. A typical scenario is an event-detection problem in the wireless sensor networks where a group of monitoring sensors try to capture few active event sources from many event resources. Specifically, we focus on two issues of wireless sensor networks. First, compared with the total number of sensors in the network, the number of active sensors is very limited. In addition, the number of events is much smaller than the number of all sources. Second, different events may occur at the same time and cause interference to detect them separately. As a result, the received signals are all superimposed, and an effective algorithm is needed to separate the signal $\mathbf{x} \in \mathbb{R}^n$.

Generally, consider the estimation of an unknown real-valued signal vector $\mathbf{x} \in \mathbb{R}^n$ from a vector of linear measurement $\mathbf{y} \in \mathbb{C}^m$ ($m < n$), i.e.,

$$\mathbf{y} = H\mathbf{x}, \quad (1)$$

where $H \in \mathbb{C}^{m \times n}$ is often referred to as a measurement matrix, the number of nonzero elements of x is small, $k \ll n$, and k is known as the sparsity of x .

More specifically, we show that the sensing process in an IoT system produces an equivalent integer CS using sparse graph codes at the sink node. For this, one can devise a simple deterministic construction of the sparse measurement matrix and an efficient

integer-valued signal recovery algorithm. Furthermore, many practical IoT systems can be modeled as integer CS based on sparse measurement matrices.

Similarly to traditional CS, the two main concerns in the integer CS of IoT systems are: (1) designing the measurement matrix H and (2) recovering \mathbf{x} from the measurements \mathbf{y} by exploiting the sparsity constraint and integer property of \mathbf{x} .

In this paper, we focus on deterministic as opposed to random measurement matrices for integer CS of IoT systems. The deterministic matrices are useful since in practice, the matrix H in Equation (1) has to be a deterministic matrix. Furthermore, by designing the proper matrix, we can improve certain features on CS of IoT systems such as recovery computational complexity and compression ratio.

Specifically, we mainly study the construction of the sparse measurement matrix and the reconstruction algorithm with integer CS based on sparse-graph codes. On the one hand, the sparse measurement matrix and reconstruction algorithm based on sparse-graph codes simplifies the construction process and lowered the measurement dimension, which is convenient for practical application.

The construction of measurement matrix must be guaranteed that each measurement can uniquely reconstruct the original signal. For example, suppose $h_1 + h_2 = h_3$, where h_1, h_2, h_3 is the linear projection coefficient. If the binary signal \mathbf{x} is measured, the measurement is denoted as $h_1x_1 + h_2x_2 + h_3x_3 = h_3$. However, two possible solutions are obtained by measurement: $x_1 = x_2 = 0, x_3 = 1$ and $x_1 = x_2 = 1, x_3 = 0$. Obviously, the matrix design does not meet the uniqueness requirement. In this paper, we mainly focus on the construction of deterministic sparse matrices and their reconstruction algorithms for integer CS of IoT systems. Our idea comes from the concept of algebraic numbers and their minimum polynomials in number theory, which ensures the uniqueness of integer signal reconstruction. The main contributions of this paper are as follows:

- (1) Using the sparse-graph codes in channel coding and integer CS of IoT systems, we design a new class of deterministic sparse measurement matrices which are sufficient to guarantee the unique recovery for integer CS of IoT systems.
- (2) By exploiting the integer-valued structure of the event-detection signal and sparse measurement matrix, an efficient event-detection signal recovery algorithm is proposed. Furthermore, the asymptotic analysis is also conducted on the performance of the recovery algorithm with the density evolution method.
- (3) In order to deal with the noise, the above-mentioned measurement matrix design is extended to the noisy situation, and an effective reconstruction criterion under the noise condition is proposed.

Throughout this paper, we use the following notations. For a vector \mathbf{x} , x_i denotes the i -th entry of \mathbf{x} . For a matrix H , we use \mathbf{h}_i to denote the i -th row of matrix H , the transpose of matrix H is denoted by H^T . For any support set Γ , the cardinality of Γ is denoted as $|\Gamma|$, $\mathbf{x}_{\{\Gamma\}}$ denotes a vector of length $|\Gamma|$ taken from \mathbf{x} with support set Γ .

2. Related Work

Several approaches have used CS theory for event detection problem. For example, the multivariate Bayesian compressive sensing (BCS) approach has been adopted in [10] to settle the event detection problem in the IoT system; however, the multivariate BCS approach is known to be sensitive to noise, resulting in poor performance in low signal-to-noise ratios (SNRs). To settle this problem, Eltabie et al., in [11], proposed the use of assisted compression to model the dense IoT problem. However, massive sensor nodes must be employed in the sensing field to guarantee its performance. Generally, the event detection problem in IoT system can be modeled as an integer compressive sensing problem.

In traditional CS, the sparse signals are defined over the field of real numbers. However, in practice, the alphabet of the sparse signal can be modeled as a finite set of integers. For example, blind estimation in digital communications [12], binary sparse signals [13,14], and other counting data sets, including arrivals at a queue/server, are inherently integer-

valued. In such cases, it is called integer CS [7,15,16], where knowledge of the sparsity and the integer property can prove to be useful.

Most of the measurement matrices in the event-detection problem of the IoT system are based on randomization, such as the i.i.d Gaussian random matrix [7,17]. In general, the exact solution to the second concern is shown to have a combinatorial nature and the computational time becomes exponential [17]. However, if both the sparsity and the integer-valued features are exploited, some efficient algorithms, such as sphere decoding and CVX in convex programming, can be performed. In this case, the exact exhaustive search with the same solution for almost all the possible inputs would not be necessary [13,17].

Most of the works on event-detection algorithms of the IoT system are based on the random matrix [13,14,18]. In [13,14], the authors focused on the binary sparse signal. The problem of detecting the entries of binary signal was formulated as a Bayesian framework in [13]. M. Shirvanimoghaddam et al. proposed a binary signal-recovery approach based on analog fountain codes. In [7], the authors provided a link between superimposed codes (SC) and integer CS, where the new codes, termed as weighted superimposed codes (WSC), were proposed. The number of required signal measurements in WSC is of $\mathcal{O}(k \log n / \log k)$, which is less than the standard $\mathcal{O}(k \log(n/k))$. In [17], to reduce the computational complexity of the recovery, the authors solved the problem by minimizing the sum of weighted absolute values. However, the probability distribution defined on an alphabet should be known. In [19–22], the authors considered compressed sensing over a finite alphabet, where the elements of the measurement matrix are also in a finite alphabet. In this paper, we revisit the event-detection problem in an IoT system from the coding theory perspective and solve the problem as a decoding problem.

3. System Model and Problem Formulation

3.1. System Model

The n event sources in the IoT system, denoted as E_1, E_2, \dots, E_n , are randomly distributed in the detection area Ξ , as shown in Figure 1. Each event source E_i randomly generates an event x_i which can be detected by the sensor nodes in the area. Since only the active state of the event source is detected, the event x_i is defined as a binary signal, indicating that the event $x_i = 1$ is an active event, and the event $x_i = 0$ is an inactive event. Note that the number of simultaneous active events is k ($k \ll n$), so the vector \mathbf{x} is a n -dimensional k -sparse binary vector.

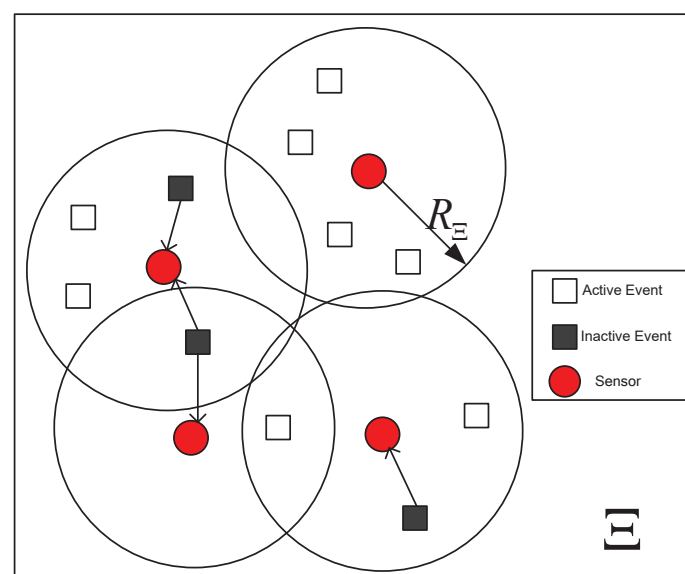


Figure 1. m active sensors are capturing k out of n active events, where R_{Ξ} is the monitoring range and the size of arrow indicates different R_{Ξ} .

Assuming that there are m sensors used to monitor the event source in the area, its monitoring range is R_{Ξ} , then each sensor can monitor the event source of the distance sensor at most R_{Ξ} . Denote $h_{i,j}$ as the channel of the i -th event source and the j -th sensor, then the signal y_j received by the j -th sensor is:

$$y_j = \sum_{i \in \Xi_j} h_{i,j} x_i + n_j, \quad (2)$$

where Ξ_j is the event source within the monitoring range of the j -th sensor, and n_j is noise. Its vector form is:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}. \quad (3)$$

Generally, sensors are randomly distributed in the monitoring area; event sources that are not within the monitoring range cannot be monitored. In other words, when the total monitoring area Ξ is large, most elements $h_{i,j}$ are zero, thus, the matrix \mathbf{H} is sparse. Therefore, when the number of sensor nodes is less than the number of event sources ($m < n$), the monitoring model of active events in the wireless sensor network can be regarded as an integer compressed sensing model based on a sparse measurement matrix. In addition, the detection of the positioning signal in GPS system can also be regarded as a sparse matrix and an integer compressed sensing model.

3.2. Problem Formulation

Generally, the event detection problem in IoT systems can be modeled as a integer CS problem. Given $k, n, q \in \mathbb{N}$ with $k \ll n$, and a finite alphabet $\mathcal{X} = \{\chi_1, \chi_2 \cdots \chi_q\} \subset \mathbb{Z}$ with $0 \in \mathcal{X}$ and $|\mathcal{X}| = q$, we consider the following ensemble:

$$\mathcal{S} = \{\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathcal{X}^n : \|\mathbf{x}\|_0 = k\}$$

This ensemble represents the space of n -dimensional signals that are k -sparse with entries coming from \mathcal{S} .

In [17], the authors discussed the uniqueness of the solution of the discrete signal recovery. If the pair (\mathcal{X}, H) is chosen to satisfy:

$$\ker H \cap \tilde{\mathcal{X}}^n = \{\mathbf{0}\}, \quad (4)$$

where $\ker H \triangleq \{\mathbf{x} \in \mathbb{C}^n : H\mathbf{x} = \mathbf{0}\}$ and $\tilde{\mathcal{X}} \triangleq \{r_i - r_j : i, j = 1, 2, \dots, q\}$, then the linear equation associated with measurements in (1) have a unique solution.

As long as the uniqueness assumption holds, the exact solution can be obtained in a finite number of steps by enumerating all possible \mathcal{X}^n in ensemble \mathcal{S} . By denoting $\mathcal{X}^n = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_\mu\}$, we can uniquely determine the values of \mathbf{x} in a finite time by comparing the measurement vector \mathbf{y} and $H\mathbf{x}_i$ ($1 \leq i \leq \mu$). However, the computational complexity of this enumerating computation is exponential. For example, when $q = 3$ and $n = 1000$, the number of comparisons is $\mu = 3^{1000}$ in the worst case.

To reduce the computational complexity, we borrow the insights of sparse graph codes in channel coding and CS [23,24]. Our basic design is that if each row of H , say \mathbf{h}_l , is sparse (most of the elements in \mathbf{h}_l are zero), then each measurement $\mathbf{h}_l \mathbf{x}$ may only involve few nonzero elements of \mathbf{x} since most of elements of \mathbf{x} are also zero; thus, the size of the problem is reduced significantly, and the few nonzero elements of \mathbf{x} can be recovered by the efficient enumerating method in an iterative manner.

4. Event Detection Using Sparse Graph Codes

This section first briefly presents a bipartite graph. Then, we explain that the event-detection problem in the IoT system is a special case of sparse graph decoding, so it can be solved with fewer measurements, by the standard peeling decoding algorithm, compared with the existing methods. We further design the measurement matrix and obtain the

lower bound of the minimum number of sensor nodes required to successfully capture all active events.

4.1. Preliminaries

Any sparse matrix can be efficiently represented by a bipartite graph. Let $G = (\mathcal{V} \cup \mathcal{C}, \partial)$ denote a bipartite graph with variable nodes set \mathcal{V} , check nodes set \mathcal{C} , and the edge set ∂ ; every edge in ∂ connects a node v_i in \mathcal{V} to a node c_i in \mathcal{C} . Additionally, let \tilde{H} denote the bi-adjacency matrix of bipartite graph G :

$$\tilde{H}_{i,j} = \begin{cases} 1, & \text{if } \{v_i, c_j\} \in \partial \\ 0, & \text{if } \{v_i, c_j\} \notin \partial \end{cases}$$

In a bipartite graph, a node in $\mathcal{C}(\mathcal{V})$ has degree i when it connects to i nodes in $\mathcal{V}(\mathcal{C})$. Let λ_i denote the fraction of nodes with degree i in \mathcal{V} , ρ_j denote the fraction of nodes with degree j in \mathcal{C} , and let

$$\lambda(x) = \sum_i \lambda_i x^{i-1}, \rho(x) = \sum_j \rho_j x^{j-1}$$

represent the degree distributions.

When the measurement matrix is sparse, the event detection problem in the IoT system also can be represented by a bipartite graph, where the sets of signal coefficients and measurements are mapped to the variable nodes set \mathcal{V} and check nodes set \mathcal{C} , respectively. In this paper, we denote the set of variable nodes that incident to check node l by $\mathcal{N}(l) = \{i : h_{l,i} \neq 0\}$.

Since the event source generates integer-value signals, the event vector can be regarded as integer-value symbols during sparse graph coding. During sensing, the vector is multiplied by the channel matrix H and sent to the sink node for further processing. Since we assume that each sensor node only receives signals from event sources in the sensor coverage area, the nonzero entries in each row of H are very small. This is very similar to the sparse graph coding process, where the generator matrix is now the channel matrix and the weight coefficient is the channel gain. More specifically, the degree of each coding symbol (measurement or sensor) is represented by the sensor degree, which is now the number of event sources. In addition, the number of nonzero elements in the corresponding row of the generator (or measurement) matrix is determined by these sources. By treating each event source (information symbol) and sensor reading (coding symbol) as variables and check nodes, respectively, the measurement process in the IoT system can be represented by a weighted bipartite graph. The construction of measurement matrix is given as follows.

4.2. Measurement Matrix Construction

We construct the measurement matrix using the row tensor product given in [25].

Definition 1 (row tensor product). Let A be a $t_1 \times n$ matrix and B a $t_2 \times n$ matrix, with rows $\{A_i : 1 \leq i \leq t_1\}$ and $\{B_j : 1 \leq j \leq t_2\}$, respectively. The row tensor product $S = A \otimes_r B$ of A and B is a $t_1 t_2 \times n$ matrix whose rows are $\{A_i B_j : 1 \leq i \leq t_1, 1 \leq j \leq t_2\}$, where $A_i B_j$ denotes the component wise product of two vectors of length n .

The measurement matrix is generated from the sparse bi-adjacency matrix \tilde{H} of the bipartite graph G . The main focus of this paper is on the left d -regular (all variable nodes have the same degree d) bipartite graph, i.e., $\lambda(x) = x^{d-1}$. The results, however, can be generalized to regular or irregular cases.

Definition 2 (measurement matrix). Given a $m \times n$ sparse bi-adjacency matrix \tilde{H} of left-regular bipartite graph and a complex vector $\mathbf{f} = [1, \dots, e^{j2\pi(i-1)/p}, \dots, e^{j2\pi(n-1)/p}]$, the $m \times n$ measurement matrix H is given by

$$H = \tilde{H} \otimes_r \mathbf{f}, \quad (5)$$

where $p \geq n$ is a constant prime.

To guarantee the exact recovery of all the k -sparse integer-valued signals, the measurement matrices should be designed first, i.e., satisfying Equation (4), which is able to uniquely determine the signal coefficients. For the constructed measurement matrix, the following theorem holds:

Theorem 1. For the measurement matrix H constructed as (5), the condition $\ker H \cap \tilde{\mathcal{X}}^n = \{\mathbf{0}\}$ holds for the pair (\mathcal{X}, H) .

To prove this theorem, we need some notations.

Definition 3 (algebraic number minimal polynomial). The minimal polynomial of an algebraic number ζ is the unique irreducible monic polynomial of smallest degree $q(x)$ with rational coefficients such that $q(\zeta) = 0$ and whose leading coefficient is 1.

For the algebraic number $e^{j2\pi/p}$ (p is a prime), the minimal polynomial is $\Phi_p(x) = \sum_{i=1}^p x^{(i-1)}$.

Lemma 1. For $\alpha_i \in \mathbb{Z}$, the equation $\sum_{i=1}^p \alpha_i e^{j2\pi(i-1)/p} = 0$ holds if and only if for every $\alpha_i = \alpha$ ($i = 1, 2, \dots, p$), α is a constant.

Next, we prove the Theorem 1 via Lemma 1.

Proof. Take any $\kappa \in \ker H \cap \tilde{\mathcal{X}}^n$. There must exist $\mathbf{x}', \mathbf{x}'' \in \mathcal{X}^n$ such that $\kappa = \mathbf{x}' - \mathbf{x}''$ as $\kappa \in \tilde{\mathcal{X}}^n$. With the definition of $\ker H$, then we have $H\kappa = H(\mathbf{x}' - \mathbf{x}'') = \mathbf{0}$, i.e., $\mathbf{h}_l(\mathbf{x}' - \mathbf{x}'') = 0$ ($1 \leq l \leq m$). From the construction as in (5), we have

$$\sum_{i \in \mathcal{N}(l)} \alpha_i e^{j2\pi(i-1)/p} + \sum_{i \notin \mathcal{N}(l)} 0 \cdot e^{j2\pi(i-1)/p} = 0, (1 \leq i \leq p), \quad (6)$$

where $\alpha_i = x'_i - x''_i$ ($i \in \mathcal{N}(l)$) is an integer. From Lemma 1, the Equation (6) holds if and only if $\alpha_i = 0$, i.e., $x'_i = x''_i$ for all $i \in \mathcal{N}(l)$. As $\bigcup_{l=1}^m \mathcal{N}(l) = \{1, 2, \dots, n\}$, we have $\mathbf{x}' = \mathbf{x}''$. Thus $\kappa = \mathbf{x}' - \mathbf{x}'' = \mathbf{0}$, this implies $\ker H \cap \tilde{\mathcal{X}}^n = \{\mathbf{0}\}$. \square

4.3. Integer Sum-Peeling Algorithm for Event Detection

The number of nonzero variable nodes contained in a measurement is called the measurement degree η and denotes $\mathcal{X}^* = \mathcal{X} \setminus \{0\}$. Then, the variable nodes connected to the measurement with $\eta = D$ ($D = 0, 1$) can be verified through the following rules.

R1. If $\|y_i\|^2 = 0$, then all the variable nodes connected to y_i are verified as having a zero value.

R2. If $\|y_i\|^2 = \|\chi_j\|^2$ and $l = (\angle \chi_j y_i / 2\pi)p$ is an integer, where $\chi_j \in \mathcal{X}^*$, $1 \leq j \leq q-1$, then the variable node l can be verified with χ_j , and all the other variable nodes connected to y_i are verified with zero values.

Furthermore, for the nonzero measurement with degree $\eta \geq D$ ($D \geq 2$), the variable nodes connected to it are verified by the enumeration method.

R3. If $y_i = \Lambda_l \mathbf{f}_{\{\mathcal{N}(i)\}}$, where Λ_l is a sparse integer vector of dimension $|\mathcal{N}(i)|$ with greater than 1 but less than D different nonzero elements χ_j ($\chi_j \in \mathcal{X}^*$, $1 \leq j \leq q-1$), then all the variable nodes connected to y_i are verified as Λ_l .

R1–R3 can be performed since the construction (5) guarantees unique solution. The parameter D is called as selective degree. Then, we propose a peeling algorithm called the integer sum-peeling (ISP) algorithm, which is presented in Algorithm 1.

Algorithm 1 Integer sum-peeling-algorithm

Input: observation vector \mathbf{y} , the measurement matrix H , the selective degree D , constant prime p .

(1) If $\|y_i\|^2 = 0$ for $1 \leq i \leq m$, verify: $\hat{x}_{\{\mathcal{N}(i)\}} = 0$.

(2) If $\|y_i\|^2 = \|\chi_j\|^2$ and $l = (\angle \chi_j y_i / 2\pi)p$ is an integer for $1 \leq i \leq m$ ($\chi_j \in \mathcal{X}^*$, $1 \leq j \leq q - 1$). Verify: $\hat{x}_l = \chi_j$, $\hat{x}_{\{\mathcal{N}(i) \setminus l\}} = 0$.

(3) If $0 < \|y_i\|^2 \neq \|\chi_j\|^2$ for $1 \leq i \leq m$, $\chi_j \in \mathcal{X}^*$, $1 \leq j \leq q - 1$ and $D > 1$, set $\tilde{\mathbf{f}} = [f_{\{\mathcal{N}(i)\}}]'$, $\bar{\rho} = |\mathcal{N}(i)|$. Construct matrix Λ , its rows consist of all sparse integer vectors of dimension $\bar{\rho}$ with greater than 1 but less than D different nonzero elements χ_j . Compute the vector: $\mathbf{1} = \Lambda \tilde{\mathbf{f}}$. Go to step 4).

(4) If $\exists l$ s.t. $\zeta_l = y_i$, verify: $\hat{x}_{\{\mathcal{N}(i)\}} = \Lambda_l$.

(5) Peel off: remove all the edges connected to verified variable nodes $\hat{x}_{\{\mathcal{N}(i)\}}$, $\mathbf{y} = \mathbf{y} - H\hat{\mathbf{x}}$.

(6) Repeat step (1)–(5) until all the variable nodes are verified.

output: the recovered sparse integer-valued signal $\hat{\mathbf{x}}$.

5. Performance Analysis of Integer Sum-Peeling Algorithm

This section elaborates on the overall performance of the ISP algorithm using density evolution, which converges to the performance of the tree-like neighborhood $G_v^{2\ell}$ as n tends to infinity [26–29]. $G_v^{2\ell}$ is defined as the subgraph containing the variable node v and all those nodes that are incident to v with any path of length less than or equal to 2ℓ .

Definition 4. A stopping set \mathcal{S} is a subset of the set of variable nodes \mathcal{V} , by which all neighbors of \mathcal{S} are connected to \mathcal{S} at least twice.

5.1. Asymptotic Analysis of Integer Sum-Peeling Algorithm

In the following, we provide a definition of the asymptotic analysis, highlighting the technical components.

Density evolution: We analyze the performance of our proposed ISP algorithm over a typical graph (i.e., cycle-free) of the ensemble for a fixed number of peeling iterations i . We assume that a local neighborhood of every edge in the graph is cycle-free (tree-like), and derive a recursive equation that represents the average density of remaining edges in the pruned graph at iteration.

In this paper, the bipartite graphs are constructed randomly with $\lambda(x) = x^{d-1}$, i.e., every variable node is connected to d check nodes, one uniformly random check node in each of the d sets. To analyze the performance of the peeling decoder performed on a tree-like graph, we need to determine another degree distribution $\rho(x) = \sum_j \rho_j x^{j-1}$, i.e., ρ_j . According to the ensemble of left d -regular bipartite graph, the degree of a check node is a binomial $B(d/m, n)$ and can be well approximated by a Poisson random variable

$$\rho_j \approx \frac{(d/\beta)^j e^{-d/\beta}}{j!}, \quad (7)$$

where $\beta = m/n$ is the sampling ratio.

With the tree-like neighborhood, the ISP procedures over different nodes in the first l iterations are independent. The ISP algorithm can decode up to D nonzero variable nodes, which can be verified by the $(l + 1)$ -th iteration if the check node $D - 1$ are connected at most. Let p_l be the expected rate of unverified nonzero variable nodes at l -th iteration ($p_0 = k/n$), then, the probability p_{l+1} can be written with regard to the probability p_l at the l -th iteration in a recursive manner:

$$p_{l+1} = p_0 \left(1 - \sum_{i=0}^{D-1} \sum_j \rho_j \binom{j-1}{i} p_l^i (1-p_l)^{j-1-i} \right)^{d-1}. \quad (8)$$

More specifically, in this paper we only consider $D = 1$, and 2. When $D = 1$, the term $\sum_j \rho_j (1 - p_l)^{j-1}$ in (8) can be approximated by e^x of Taylor expansion

$$\rho(x) = \sum_j \rho_j x^{j-1} = e^{-(1-x)\frac{d}{\beta}} / x. \quad (9)$$

Therefore, the density evolution Equation (8) can be obtained as

$$p_{l+1} = p_0 \left(1 - \frac{e^{-\frac{d}{\beta} p_l}}{1 - p_l}\right)^{d-1}. \quad (10)$$

Similarly, when $D = 2$, the density evolution equation can be given by

$$p_{l+1} \approx p_0 \left(1 - \frac{e^{-\frac{d}{\beta} p_l}}{1 - p_l} - p_l \frac{e^{-\frac{d}{\beta} p_l}}{1 - p_l} \frac{d}{\beta}\right)^{d-1}. \quad (11)$$

Table 1 lists the analytical success thresholds of the ISP algorithm for measurement matrix with different d , β and D . As can be seen, the success thresholds improve with increased D . Furthermore, when fixing the sampling ratio β , as shown in Table 1, the overall performance of the ISP algorithm is improved with decreased d .

Table 1. Success thresholds of ISP for different (d, β) .

(d, β)	(3, 0.1)	(3, 0.2)	(4, 0.1)	(4, 0.2)	(6, 0.1)	(6, 0.2)
$D = 1$	0.0846	0.1758	0.0792	0.1629	0.0647	0.1319
$D = 2$	0.1627	0.3444	0.1378	0.2867	0.1043	0.2136

Our next set of results investigate the degree of agreement between simulation results of ISP algorithm and asymptotic analysis, the plots in Figures 2 and 3 correspond to the evolution of p_l with iterations l for the ISP algorithm over the $(d = 3, \beta = 0.2)$. When $D = 1$ and $(d = 3, \beta = 0.2)$, the success threshold is 0.1758. It can be observed that the case of $k/n = 0.1$ (below the success threshold) tends to 0 for smaller values of l compared with $k/n = 0.17$ (near the success threshold), while the in the case of $k/n = 0.2$ (above the success threshold) no more tend to 0. The simulation results of $D = 2$ are also very similar to $D = 1$, as shown in Figure 3. We observe that the two sets of results are in close agreement, particularly with the asymptotic analysis.

Notably, the above analysis is based on the assumption of a tree-like neighborhood. This is true when n, k goes to infinite. The concentration results and convergence to a tree-like case have been investigated widely in channel coding and CS [26,30], and our results in this regard are also consistent with [30], so, this content is omitted due to the limited space.

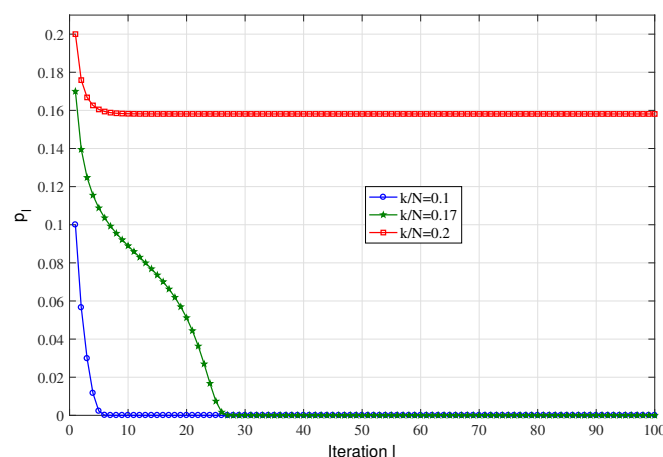


Figure 2. Evolution of p_l vs. l for the ISP algorithm over $(d = 3, \beta = 0.2)$ and $D = 1$.

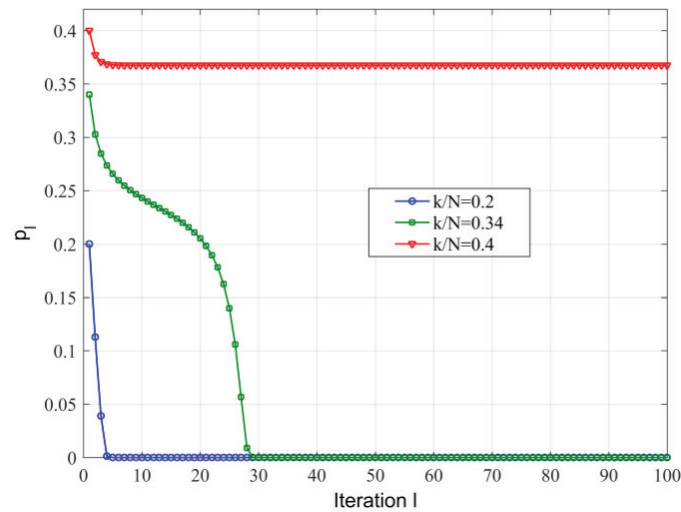


Figure 3. Evolution of p_l vs. l for the ISP algorithm over $(d = 3, \beta = 0.2)$ and $D = 2$.

5.2. Finite Length Analysis of Integer Sum-Peeling Algorithm

This section evaluates the performance of an ISP algorithm with finite n . This is followed by a stopping set analysis of the decoding algorithm in binary erasure channel (BEC), but with some differences. In each iteration, the ISP algorithm can verify up to D nonzero variable nodes from a check node, while the traditional peeling algorithm can only decode, at most, one erasure in BEC. In order to exploit the similarity between the ISP algorithm and the traditional algorithm used on BEC, we expand Definition 4 to a generalized stopping sets.

Definition 5 (D -stopping set). A D -stopping set \mathcal{S}_D is a subset of the set of variable nodes \mathcal{V} such that all neighbors of \mathcal{S}_D are connected to \mathcal{S}_D at least $D + 1$ times.

The next lemma shows the key role that D -stopping set play in the ISP algorithm.

Lemma 2. Let regular (λ, ρ) -H constructed in (5) be a measurement matrix. Let \mathcal{T} denote the set of nonzero coefficients of sparse binary signal. Then the set of nonzero variable nodes which remains when the sum-peeling algorithm stops is equal to the unique maximal D -stopping set of \mathcal{T} .

Proof. Let \mathcal{S}_D be a D -stopping set included in \mathcal{T} . Consider a special case when all other coefficients are known, every neighbor of \mathcal{S}_D has at least $D + 1$ connections to the set \mathcal{S}_D so that the ISP algorithm can not verify the nonzero variable nodes included in the unique maximal stopping set contained in \mathcal{T} . Conversely, if the ISP algorithm stops at a set \mathcal{S}_D , then this happens only if all neighbors of \mathcal{S}_D have at least $D + 1$ connections to \mathcal{S}_D . Additionally, since zero variable nodes included in a D -stopping set can be verified by the ISP algorithm, \mathcal{S}_D must be maximal D -stopping set. \square

Lemma 2 gives a foundation to implement the theoretical analysis and exploit the similar ensemble of LDPC results in BEC [31]. In [31], the authors developed combinatorial approaches to compute recursively the exact average bit erasure probability performance of the LDPC codes. Following these combinatorial approaches, the average unverified variable node probability (equivalent to bit erasure probability in BEC) can be calculated exactly. In the simulation results, we show the average verified nonzero variable node probability as a function of density factor α for finite n .

5.3. The Number of Measurements and Computational Complexity of Integer Sum Peeling Algorithm

The number of measurements: The ISP architecture (see Definition 2 (measurement matrix) in Section 3) with d stages, has d distinct measurement patterns, chosen as per the discussions in Sections 3 and 4. These constructions are deterministic and are pre-computed. We assume that there is random access memory, and the unit cost of each I/O operation is used to read measurements. As shown in Section 4.1, we have three different ISP architectures, with $d = 3$, $d = 4$ and $d = 6$ regions, for different D . The total measurements used for these regions are $m = 1.182k$, $m = 1.2626k$, and $m = 1.5456k$, respectively, for $D = 1$ and $\beta = 0.1$. In general, for any fixed d and D , the number of measurements m can be as small as $\mathcal{O}(k)$.

It can be easily concluded that the proposed ISP requires less measurements than the existing method [14,19], which the number of required measurements is of $\mathcal{O}(-n\log(1 - k/n))$ and $\mathcal{O}(k\log(n))$. We also verify the measurement dimension required by these methods in simulations.

The computational complexity: In the proposed recovery algorithm, the maximum value of D is only considered to be 2. As shown in [24], the arithmetic operation is of $\mathcal{O}(k)$ for traditional CS. When D is set to 1, the main difference between ISP and traditional CS [24] is the rule **R2**, where the ISP algorithm requires extra $q - 1$ comparisons. Thus, the recovery complexity is of $\mathcal{O}(k(q - 1))$.

It also can be easily concluded that the proposed ISP requires less iterations than the existing method [14,19], which the number of required iterations is of $\mathcal{O}(-n\log(1 - k/n))$ and $\mathcal{O}(k\log(n))$.

6. Noisy Measurements

We consider the following model of measurements contaminated with noise:

$$\mathbf{y} = H\mathbf{x} + \mathbf{n}, \quad (12)$$

where the measurements \mathbf{y} of (1) were corrupted by an additive noise vector \mathbf{n} . The noise vector to the measurements results in the probability of having a 0 or integer amplitude of measurement to be zero. This will disable the rules **R1**, **R2** and **R3** in verifying the variable nodes. However, the rules can be effective by slight refinements.

We firstly show how to generalize the measurement matrix H of (5) in the noisy case. Following with (5), the $m \times n$ measurement matrix is given by

$$H = \begin{bmatrix} \tilde{\mathbf{h}}_1 \otimes F \\ \vdots \\ \tilde{\mathbf{h}}_{m'} \otimes F \end{bmatrix}, \quad (13)$$

where the $T \times n$ matrix F is

$$F = \begin{bmatrix} 1 & \cdots & 1 & \cdots \\ 1 & \cdots & W^{i-1} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \cdots & W^{(i-1)(T-1)} & \cdots \end{bmatrix} \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ 0 & g_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & g_n \end{bmatrix},$$

$m = Tm'$, $W = e^{-j2\pi/p}$ ($p \geq n$ is a prime), $\tilde{\mathbf{h}}_i$ be the i -th row of the $m' \times n$ local matrix \tilde{H} and the weight entry g_i is from i.i.d Gaussian distribution. The main difference between the construction of (5) and (13) is single vector \mathbf{f} to matrix F , this results in single measurement y_i to T measurements y_i, \dots, y_{i+T-1} (denotes as \mathbf{y}_T vector). Next, we exploit a similar rule of a noiseless case based on \mathbf{y}_T .

R4. If $\|\mathbf{y}_T - F_{\{\mathcal{N}(i)\}}\Lambda_i\|^2 \leq (1 + \varepsilon)\sigma^2$, where $F_{\{\mathcal{N}(i)\}}$ is taken from the columns of matrix F with support set $\mathcal{N}(i)$, Λ_i is a binary vector of dimension $|\mathcal{N}(i)|$ with at most D ones, σ^2 is variance of noise vector and ε is a parameter between 0 and 1. Then, all the variable nodes connected to \mathbf{y}_T are verified as Λ_i .

Similarly, the rule **R4** is responsible for verifying the variable nodes via the peeling-decoding algorithm. Therefore, the reconstruction algorithm in a noisy case is essentially identical with the noiseless case, i.e., the ISP algorithm. We will show the validity of the ISP algorithm with **R4** in the next section.

7. Simulation Results

In this section, we evaluate the performance of ISP and conventional sparse event detection algorithms through numerical simulations. We also show the degree of agreement between the asymptotic analysis and simulation results. In all simulations, the left d -regular bipartite graph $G(d, \beta)$ ($\beta = m/n$) is constructed randomly.

Firstly, we compare the proposed ISP algorithm with conventional basis pursuit (BP) algorithm in CS [1] and the sum of absolute values (SAV) algorithm [17]. Three event sources cases of \mathcal{X} are considered (same with [17]): $\mathcal{X}_1 = \{0, 1\}$, $\mathbb{P}(0) = \text{Pr}$, $\mathbb{P}(1) = 1 - \text{Pr}$; $\mathcal{X}_2 = \{-1, 0, 1\}$, $\mathbb{P}(0) = \text{Pr}$, $\mathbb{P}(1) = \mathbb{P}(-1) = (1 - \text{Pr})/2$; $\mathcal{X}_3 = \{-2, -1, 0, 1, 2\}$, $\mathbb{P}(0) = \text{Pr}$, $\mathbb{P}(-2) = \mathbb{P}(-1) = \mathbb{P}(1) = \mathbb{P}(2) = (1 - \text{Pr})/4$. In [17], the author considers the finite alphabet set \mathcal{X} that does not necessarily contain 0, i.e., $0 \leq \text{Pr} \leq 1$. In this paper, we only consider the sparse integer-valued signal with high Pr . In the SAV and BP algorithms, the simulations are based on randomly generated measurement matrix $H \in \mathbb{R}^{200 \times 1000}$ whose entries are chosen independently from standard Gaussian distribution. In the proposed ISP algorithm, the measurement matrix $H \in \mathbb{C}^{100 \times 1000}$ is adopted as constructed in (5) with the bipartite graph $G(3, 0.1)$.

Figure 4 illustrates the averaged NSR (noise-to-signal ratio) $\|\mathbf{x} - \hat{\mathbf{x}}\|_2 / \|\mathbf{x}\|_2$ as a function of the probability Pr of zero elements in \mathcal{X} . Note that the lower critical probability Pr implies a better recovery performance. As shown in the figure, the critical probability Pr of the proposed ISP algorithm ($D = 1$ and $D = 2$) is lower than the BP and SAV algorithms for all cases of \mathcal{X} . Furthermore, the performance of the proposed ISP algorithm is independent of the number of symbols q , while the BP and SAV algorithms have worse performance, with decreased q .

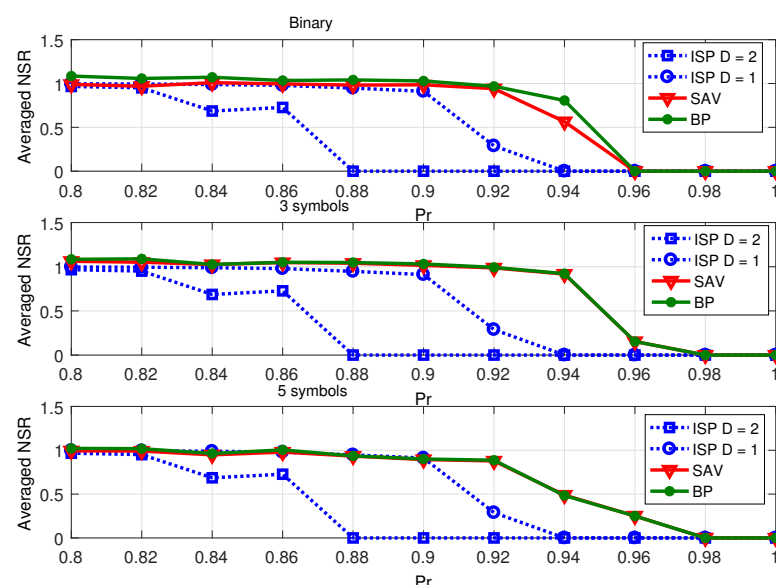


Figure 4. Averaged NSR performance of sparse recovery algorithms for integer-valued signal.

Secondly, we show the number of measurements (the number of sensors) required to exactly detect the sparse events versus the sparsity order. Each nonzero element of \mathbf{x} is randomly chosen from the event sources set $\mathcal{X}^* = \{+1, -1\}$. The support of \mathbf{x} (the position of active events) is chosen randomly, $d = 3$ is adopted, the $n = 10^4$ event sources are randomly distributed in an area of 1000 m by 1000 m, and the propagation loss factor is 3. As shown in Figure 5, the number of sensors of the proposed ISP method is significantly lower than that for the asymptotic bound of the WSC method [7] and standard CS ($D = 1$ and $D = 2$). In addition, as seen in the figure, the simulation results are close with the asymptotic threshold.

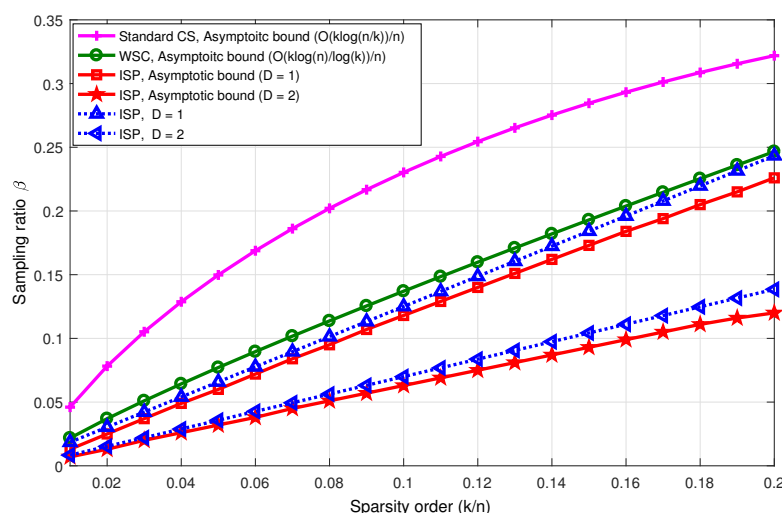


Figure 5. The minimum sampling ratio required to exactly recover the integer-valued sparse signal of dimension $n = 10^4$ versus the sparsity order.

Thirdly, we apply the ISP algorithm to six randomly constructed left d -regular bipartite graphs $G(d = 6, \beta = 0.05)$ with event sources $n = 3 \times 10^3, 10^4, 10^5$ to investigate the agreement between our asymptotic analysis and simulation results. Each nonzero element of \mathbf{x} is randomly chosen from the set $\mathcal{X}^* = \{+1, -1\}$. Let p_0 denote the maximum sparsity order for the signal of length n , to be exactly recovered by the proposed ISP algorithm. The exact recovery of thresholds p_0 are listed in Table 2. As can be seen, for all graphs, the larger n indicates higher closeness between the thresholds p_0 and the asymptotic threshold. Accordingly, the simulation results match well with the theoretical results.

Table 2. Asymptotic and simulation success thresholds for ISP.

D	3×10^3	1×10^4	1×10^5	Asymptotic
1	$p_0 = 0.0013$	$p_0 = 0.0023$	$p_0 = 0.003$	$p_0 = 0.0031$
2	$p_0 = 0.0027$	$p_0 = 0.0038$	$p_0 = 0.0047$	$p_0 = 0.0051$

Fourthly, we compare the performance of the ISP algorithm with that of existing binary event detection algorithms considering binary event detection, a special class of event sources.

- (1) Consider that the $n = 500$ event sources are randomly distributed in an area of 500 m by 500 m, the propagation loss factor is 3 and the sparsity k is set to 100. The optimal row weights of the sparse matrix corresponding to AFCS are set, respectively, when $D = 1, L = 7$, and when $D = 2, L = 9$. ISP and BCS also employ the sparse matrices of the AFCS scheme for a fair comparison. Figure 6 shows the reconstruction error rate performance curves of the ISP algorithm and the two existing reconstruction algorithms of AFCS and BCS with the measurement dimension varying. The re-

construction error rate is defined as the ratio of the number of nonzero signals that were not successfully reconstructed to the signal dimension. From Figure 6, it can be observed that the performance of the proposed ISP algorithm is basically the same as that of the AFCS algorithm; however, previous analysis reveals the lower complexity of the ISP algorithm compared with the AFCS algorithm. On the other hand, it can also be observed that both the ISP and AFCS reconstruction algorithms outperform the BCS scheme.

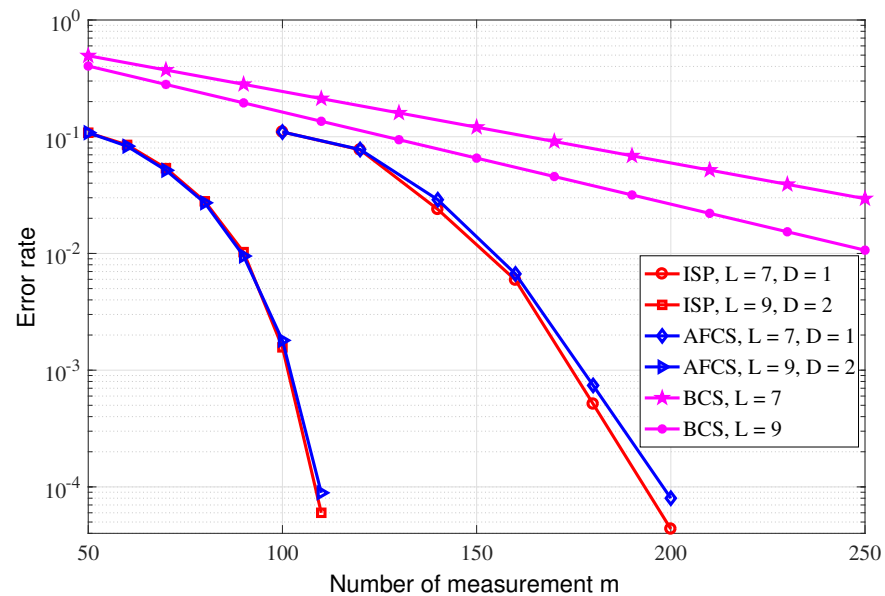


Figure 6. Performance comparison between ISP and other existing binary signal reconstruction algorithms.

- (2) In the AFCS algorithm, the optimal row weight of the sparse matrix is $L = \lceil -0.5(D + 2)/\log(1 - k/n) \rceil$; however, the AFCS scheme generally cannot optimize the row weight of its measurement matrix because of the actual unknown or variable sparsity. In this experiment, we fix the row weight of the sparse matrix, and compare the performance of the ISP and AFCS algorithms. Consider two experimental situations, $L = 40$, event sources $n = 1000$ and $L = 120$, event sources $n = 2000$. Figure 7 reveals the better performance of the ISP algorithm compared with the AFCS algorithm, for a fixed row weight, which suggests that the ISP scheme is robust to sparsity varying. In addition, it can also be observed that there is a broken line in AFCS. This is because the AFCS constructs the measurement matrix adaptively, and when the row weight is not optimal, there may be no solution.
- (3) The theoretical performance of the above three event detection algorithms is also compared. In the simulation, event sources $n = 10^5$ are randomly distributed in an area of 1000 m by 1000 m, the propagation loss factor is 3. The measurement matrix is a randomly generated left-regular sparse matrix, where the column weight is $d = 3$. According to [14], it can be observed that the measurement dimension required by AFCS to accurately reconstruct sparse binary signals is $\mathcal{O}(-n \log(1 - k/n))$, which is smaller than $\mathcal{O}(k \log(n))$ that required by BCS. Figure 8 demonstrates the lower sampling rate required for accurate reconstruction of the ISP algorithm proposed in this paper compared with the BCS and AFCS schemes when $D = 2$, and the performance of the ISP algorithm and AFCS algorithm are basically identical when $D = 1$. Furthermore, it can be found that the simulation results of the ISP reconstruction algorithm are basically consistent with the theoretical results.

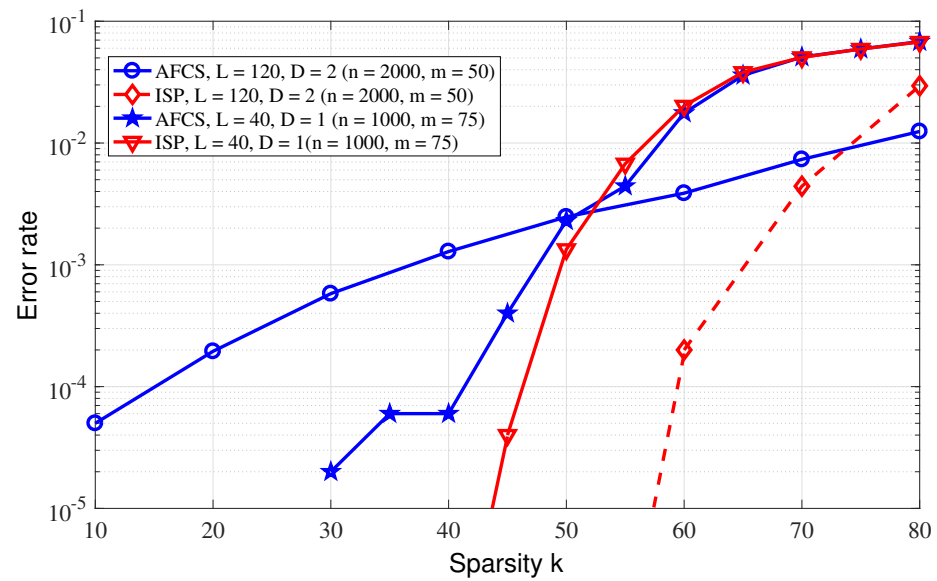


Figure 7. Performance comparison between ISP and AFCS reconstruction algorithm with fixed row weight.

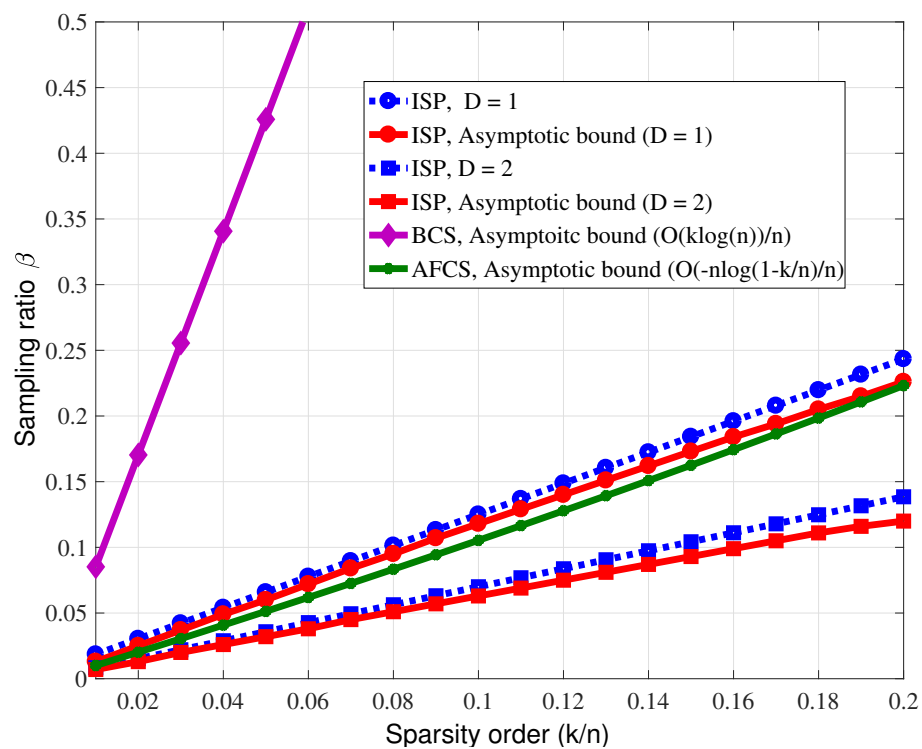


Figure 8. Minimum sampling rate required for accurate reconstruction of binary sparse signals.

Finally, we show the validity and performance of the ISP algorithm in a noisy environment. The binary sparse event sources are adopted for ease of comparison. The signal-to-noise ratio (SNR) is defined as:

$$\text{SNR} = 10 \log_{10} \frac{\|\mathbf{H}\mathbf{x}\|^2}{\|\mathbf{n}\|^2} \text{ dB}, \quad (14)$$

Figure 9 illustrates the MSE performance of the ISP and binary ℓ_1 reconstruction algorithms as a function of SNR. For the binary ℓ_1 reconstruction algorithm, a standard Gaussian random matrix is taken, and for the ISP algorithm, the rows of the sparse matrix

are reset to 3, $T = 4$. In all simulation experiments, event sources $n = 10^3$ are randomly distributed in an area of 500 m by 500 m, the propagation loss factor is 3, the matrix measurement dimension is 100×1000 , and the sparsity is set to $k = 20$. According to the figure, it can be observed that the MSE performance of the ISP algorithm is better than that of the event detection algorithm when $D = 2$. This also means that the ISP algorithm is also effective in noisy environments.

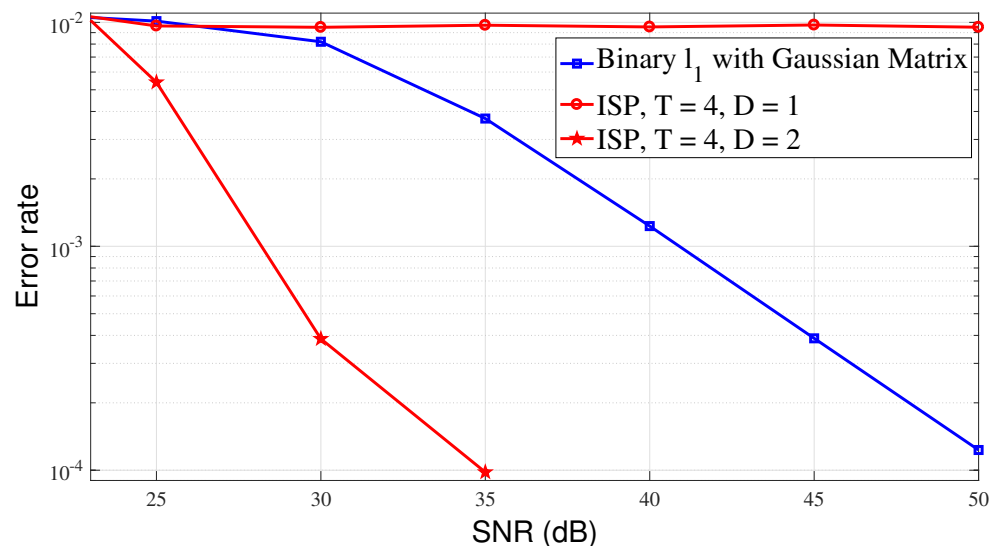


Figure 9. MSE performance under ISP and l_1 reconstruction algorithm.

8. Conclusions

In this paper, the event-detection problem is modeled as the decoding of sparse graph code, and the problem in the IoT system is described from the perspective of coding theory. Using compressive sensing, the high-dimensional event sources are recovered from incomplete linear measurements. A simple deterministic sparse measurement matrix was adopted to uniquely determine the event sources, based on which an integer sum-peeling recovery algorithm was proposed to exploit the structure of event detection. To corroborate the performance of the recovery algorithm, the asymptotic analysis was provided via the density evolution, which was verified by simulation. Simulation results also showed that the event detection algorithm had superior performance compared with other existing algorithms.

Author Contributions: J.C. (Jun Cai); Data curation, J.C. (Jun Cai); Formal analysis, J.C. and H.Z.; Funding acquisition, X.X.; Investigation, J.C. (Jian Cheng); Project administration, X.X.; Resources, X.X. and H.Z.; Software, X.X. and H.Z.; Visualization, J.C. (Jian Cheng); Writing—original draft, J.C. (Jun Cai); Writing—review & editing, X.X. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the National Natural Science Foundation of China (No.61802425, 62171466 and 62001515).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank editors and three anonymous reviewers for detailed comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Donoho, D.L. Compressed Sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [\[CrossRef\]](#)
- Candes, E.J.; Romberg, J.; Tao, T. Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information. *IEEE Trans. Inf. Theory* **2004**, *52*, 489–509. [\[CrossRef\]](#)
- Yu, T.; Wang, X.; Shami, A. UAV-enabled spatial data sampling in large-scale IoT systems using denoising autoencoder neural network. *IEEE Internet Things J.* **2018**, *6*, 1856–1865. [\[CrossRef\]](#)
- Gupta, V.; De, S. An Energy-Efficient Edge Computing Framework for Decentralized Sensing in WSN-Assisted IoT. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 4811–4827. [\[CrossRef\]](#)
- Zhao, S.; Qi, G.; He, T.; Chen, J.; Liu, Z.; Wei, K. A Survey of Sparse Mobile Crowdsensing: Developments and Opportunities. *IEEE Open J. Comput. Soc.* **2022**, *3*, 73–85. [\[CrossRef\]](#)
- Sekar, K.; Devi, K.S.; Srinivasan, P. Compressed Tensor Completion: A Robust Technique for Fast and Efficient Data Reconstruction in Wireless Sensor Networks. *IEEE Sens. J.* **2022**, *22*, 10794–10807. [\[CrossRef\]](#)
- Dai, W.; Milenkovic, O. Weighted superimposed codes and constrained integer compressed sensing. *IEEE Trans. Inf. Theory* **2009**, *55*, 2215–2229. [\[CrossRef\]](#)
- Charbiwala, Z.; Kim, Y.; Zahedi, S.; Friedman, J.; Srivastava, M.B. Energy efficient sampling for event detection in wireless sensor networks. In Proceedings of the 2009 ACM/IEEE International Symposium on Low Power Electronics and Design, San Francisco, CA, USA, 19–21 August 2009; pp. 419–424.
- Stecklein, M.; Beytur, H.B.; de Veciana, G.; Vikalo, H. Optimizing resource constrained distributed collaborative sensing. In Proceedings of the 2021 IEEE International Conference on Communications Workshops (ICC Workshops), Montreal, QC, Canada, 14–23 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
- Hwang, S.; Ran, R.; Yang, J.; Kim, D.K. Multivariate Bayesian compressive sensing in wireless sensor networks. *IEEE Sens. J.* **2015**, *16*, 2196–2206. [\[CrossRef\]](#)
- Eltabie, O.M.; Ghuniem, A.M.; Abdelkader, M.F. Model assisted compressive data gathering in dense IoT monitoring of water distribution networks. In Proceedings of the 2019 IEEE International Smart Cities Conference (ISC2), Casablanca, Morocco, 14–17 October 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 444–450.
- Van der Veen, A.J.; Talwar, S.; Paulraj, A. Blind estimation of multiple digital signals transmitted over FIR channels. *Signal Process. Lett. IEEE* **1995**, *2*, 99–102. [\[CrossRef\]](#)
- Tian, Z.; Leus, G.; Lottici, V. Detection of sparse signals under finite-alphabet constraints. In Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Taipei, Taiwan, 19–24 April 2009; pp. 2349–2352.
- Shirvanimoghaddam, M.; Li, Y.; Vucetic, B.; Yuan, J.; Zhang, P. Binary Compressive Sensing via Analog Fountain Coding. *IEEE Trans. Signal Process.* **2015**, *63*, 6540–6552. [\[CrossRef\]](#)
- Baraniuk, R.G.; Cevher, V.; Duarte, M.F.; Hegde, C. Model-based compressive sensing. *IEEE Trans. Inf. Theory* **2010**, *56*, 1982–2001. [\[CrossRef\]](#)
- Sarangi, P.; Pal, P. Measurement Matrix Design for Sample-efficient Binary Compressed Sensing. *IEEE Signal Process. Lett.* **2022**, *29*, 1307–1311. [\[CrossRef\]](#)
- Nagahara, M. Discrete Signal Reconstruction by Sum of Absolute Values. *Signal Process. Lett. IEEE* **2015**, *22*, 1575–1579. [\[CrossRef\]](#)
- Kota, P.K.; LeJeune, D.; Drezek, R.A.; Baraniuk, R.G. Extreme Compressed Sensing of Poisson Rates from Multiple Measurements. *IEEE Trans. Signal Process.* **2022**, *70*, 2388–2401. [\[CrossRef\]](#)
- Das, A.K.; Vishwanath, S. On finite alphabet compressive sensing. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Vancouver, BC, Canada, 26–31 May 2013; pp. 5890–5894.
- Bioglio, V.; Coluccia, G.; Magli, E. Sparse image recovery using compressed sensing over finite alphabets. In Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 1287–1291.
- Iimori, H.; De Abreu, G.T.F.; Hara, T.; Ishibashi, K.; Stoica, R.A.; González, D.; Gonsa, O. Robust symbol detection in large-scale overloaded NOMA systems. *IEEE Open J. Commun. Soc.* **2021**, *2*, 512–533. [\[CrossRef\]](#)
- Fosson, S.M. Non-convex Lasso-kind approach to compressed sensing for finite-valued signals. *arXiv* **2018**, arXiv:1811.03864.
- Luby, M.G.; Mitzenmacher, M.; Shokrollahi, M.A.; Spielman, D. Efficient erasure correcting codes. *IEEE Trans. Inf. Theory* **2001**, *47*, 569–584. [\[CrossRef\]](#)
- Li, X.; Pawar, S.; Ramchandran, K. Sub-linear time support recovery for compressed sensing using sparse-graph codes. *arXiv* **2014**, arXiv:1412.7646.
- Gilbert, A.C.; Strauss, M.J.; Tropp, J.A.; Vershynin, R. One sketch for all: Fast algorithms for compressed sensing. In Proceedings of the ACM Symposium on Theory of Computing, San Diego, CA, USA, 11–13 June 2007; ACM: New York, NY, USA, 2007; pp. 237–246.
- Richardson, T.J.; Urbanke, R.L. The capacity of low-density parity-check codes under message-passing decoding. *IEEE Trans. Inf. Theory* **2001**, *47*, 599–618. [\[CrossRef\]](#)
- Fang, Y.; Liew, S.C.; Wang, T. Design of distributed protograph LDPC codes for multi-relay coded-cooperative networks. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 7235–7251. [\[CrossRef\]](#)
- Shao, S.; Hailes, P.; Wang, T.Y.; Wu, J.Y.; Maunder, R.G.; Al-Hashimi, B.M.; Hanzo, L. Survey of turbo, LDPC, and polar decoder ASIC implementations. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2309–2333. [\[CrossRef\]](#)
- Fang, Y.; Bu, Y.; Chen, P.; Lau, F.C.; Al Otaibi, S. Irregular-mapped protograph LDPC-coded modulation: A bandwidth-efficient solution for 6G-enabled mobile networks. *IEEE Trans. Intell. Transp. Syst.* **2021**, *24*, 2060–2073. [\[CrossRef\]](#)

30. Eftekhari, Y.; Heidarzadeh, A.; Banihashemi, A.H.; Lambadaris, I. Density evolution analysis of node-based verification-based algorithms in compressed sensing. *IEEE Trans. Inf. Theory* **2012**, *58*, 6616–6645. [[CrossRef](#)]
31. Di, C.; Proietti, D.; Richardson, T.J.; Urbanke, R.L. Finite-length analysis of low-density parity-check codes on the binary erasure channel. *IEEE Trans. Inf. Theory* **2002**, *48*, 1570–1579.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.