LETTER

Sparse Recovery Using Sparse Sensing Matrix Based Finite Field Optimization in Network Coding

Ganzorig GANKHUYAG[†a)], **Eungi HONG**[†b)], ***Nonmembers*, *and*** **Yoonsik CHOE**[†c)], ***Member***

SUMMARY Network coding (NC) is considered a new paradigm for distributed networks. However, NC has an all-or-nothing property. In this paper, we propose a sparse recovery approach using sparse sensing matrix to solve the NC all-or-nothing problem over a finite field. The effectiveness of the proposed approach is evaluated based on a sensor network.

key words: *network coding, compressive sensing, sparse matrices, all-or-nothing property*

1. Introduction

In contrast to simple forwarding of data packets, network coding (NC) provides a means for intermediate nodes to combine new data with the forwarded data packets and thus improve throughput of the network system [1]. Such throughput improvement has been proven to reach the max-flow capacity of a network, and this benefit, together with the efficiency and scalability of NC, makes the method useful in many situations, particularly for distributed networks such as sensor networks and ad hoc networks. During the last decade, there have been many efforts to improve coding gain, achieve maximum network capacity, and increase the throughput and robustness when sufficient packets are received for perfect decoding. However, the all-or-nothing problem has not been well studied. In the all-or-nothing problem, the client node cannot decode any data from the received packets unless it receives at least the same number of coded packets as the original number of packets.

Some approaches to resolve the all-or-nothing problem have been proposed. An approximate decoding algorithm for sensor networks where the received packet size is less than that of the original packet has been proposed [2]. This algorithm decodes the correlated source data using additional information, such as similarity information of the original data. In addition, a rank-metric code with an NC technique for error correction has been proposed [3]. With this technique, source packets can be decoded if the number of lost packets is less than the minimum distance provided by the rank-metric code. However, previous approaches increase the packet size and complexity of intermediate nodes.

Manuscript received September 12, 2016.
Manuscript revised October 21, 2016.
Manuscript publicized November 4, 2016.

†The authors are with the Dept. of Electrical & Electronics Engineering, Yonsei University, 134 Shinchon-dong Seodaemun-gu, Seoul 120–749, Korea.
a) E-mail: gnzrg25@yonsei.ac.kr
b) E-mail: hek722@yonsei.ac.kr
c) E-mail: yschoe@yonsei.ac.kr (Corresponding author)

DOI: 10.1587/transinf.2016EDL8189

In contrast, other approaches to overcome the all-or-nothing problem by combining NC and a compressive sensing method have been proposed. Compressive sensing is a powerful method that can acquire and reconstruct a compressible signal by finding solutions for an undetermined system. There are two conditions for successful recovery, i.e., signal sparsity in some domain and a measurement matrix to satisfy the recovery condition for compressive sensing [4]. A combination of NC and compressive sensing has been proposed [5]. In this approach, the NC transfer matrices are generated with Bernoulli distribution, and correlated sensor network data are processed as real numbers with finite precision. Similarly, in another study, the original data dimensional space is reduced via random projection, and the nodes execute NC over a real field [6]. However, these combined approaches solve the all-or-nothing problem over a real field. To the best of our knowledge, if NC is performed in a real field instead of a finite field, the payload size may be extended because of the combining operation. In this paper, we propose a framework for NC based on a compressive sensing technique in the finite field that can avoid the all-or-nothing problem. At intermediate nodes, NC is performed using sparse network transfer matrices that satisfy the compressive sensing recovery condition. When packet loss occurs at the client side, the correlated data is recovered using a compressive sensing recovery algorithm in the finite field with sparse network transfer matrix.

The remainder of this paper is organized as follows. In Sect. 2, the proposed system design with correlated source data, the encoding scheme, and the compressive sensing based recovery algorithm are presented. In Sect. 3, we evaluate the performance of the proposed approach in an illustrative sensor network scenario. Conclusions are presented in Sect. 4.

2. Proposed Design

2.1 Random Linear Network Code

In our network system, sources, intermediate nodes, and clients are distributed over an ad hoc network. Let us assume a data vector $u_n(\in \mathbb{R})$ is generated by N discrete sources, where u_n for $1 \leq n \leq N$. Here, all neighboring data represent the same event or sequence of events; thus, the data are highly and spatially correlated, i.e., in the neighboring source data, u_n is k-sparse in a well-known transformation domain Ψ (i.e., wavelet and discrete cosine transform

(DCT)), such that

$$s_n = \Psi u_n, \tag{1}$$

where s_n is the k-sparse vector of u_n, the number of non-zero elements in s_n is $k \ll N$. The measured vector of s_n can be ordered in magnitude. The best k-sparse approximation keeps the k-largest coefficients and remaining as zero [5]. In practice, k-sparse approximation is a useful method to make k-sparse signal; however, it generates an approximation error. The transform domain matrix $\Psi^{N\times N} = \{\psi_1, \dots, \psi_N\}$ is a unitary matrix $\Psi\Psi^T = I$, i.e., $u_n = \Psi^T s_n$. We assume that the transform domain matrix Ψ is known by the intermediate node and client.

Before the data are combined in intermediate nodes, conversion of the source data from real field ($\mathbb{R}$) to finite field ($\mathbb{F}$) is necessary, each s_n is discretized and mapped into an element in $\mathbb{F}$. The quantization function Q[.] transforms the source data to $\mathbb{F}$ with size of 2^p. Thus, the field conversion is based on the quantization and dequantization function [4], which is expressed as:

$$\begin{cases} \mathbb{R} \to \mathbb{F}, & x^{(i)} = Q\left[s^{(i)}\right], \quad i = 1, \dots, w \\ \mathbb{F} \to \mathbb{R}, & s^{(i)} = DQ\left[x^{(i)}\right], \quad i = 1, \dots, w \end{cases}, \tag{2}$$

where w is element length of x_n and s_n, $x^{(i)}$ is the vector value in the finite field of $s^{(i)}$, mapped by $Q[.]$.

After the field conversation, a random linear network code (RLNC) is processed in the intermediate node. The packet generated from the RLNC can be expressed as

$$y = C * x, \tag{3}$$

where y is a linear combination of x and network transform matrix C in the finite field, and $*$ denotes the matrix multiplication in the finite field, i.e., y is a random projection vector of x by C. We assume that the network transform matrix C has the following properties. The sparse network transfer matrix satisfies two different properties: a non-singularity for RLNC decoding and restricted isometry property for compressive sensing recovery. Each element of c_{ij} in the sparse network transform matrix is considered an i.i.d. random variable with an associated probability mass function.

$$P(c_{ij} = k) = \begin{cases} 1 - \delta, & k = 0 \\ \delta/2^p - 1, & k \in \{1, 2, \dots, 2^p - 1\}. \end{cases} \tag{4}$$

Here, δ is the sparsity factor. If δ is low, then the probability of entry in C is close to zero. The sparsity factor δ is influences the complexity of intermediate nodes. In fact, in the experimental testbed, random linear NC with a sparse network matrix has been applied in order to minimize the encoding complexity in intermediate nodes [7]. Besides, success of the random linear network decoding is directly related to non-singularity of the network transfer matrix. In addition, random linear NC is directly related to the nonsingularity of the network transfer matrix. A previous study demonstrated that for a given sparsity factor, the probability of having a nonsingular transfer matrix is high when the field size is sufficiently large [8]. The sparsity factor is directly related to the effectiveness of the proposed approximate decoding algorithm. In our error prone network model, we assume that if ε packet loss occurs and l is the number of intermediate node, the client obtains a $C^{m\times n}$ network transfer matrix, where $m = l - \varepsilon$, m is determined by considering packet loss under given network conditions. The network transform matrix C, also referred to as a sensing matrix, ensures compressive sensing sparse recovery. Sufficient conditions for a sparse recovery solution depend on the restricted isometry property (RIP) [4]. Sufficient conditions are important in the construction of sensing matrices. The RLNC is generated considering a $C^{m\times n}$ sparse sensing matrix. To guarantee efficient recovery of the k-sparse signal such that sensing matrix C satisfies the RIP [4]. However, sparse random matrices cannot satisfy the RIP property. However, a sparse sensing matrix provides a different form of this property, i.e., RIP(p), which is expressed as

$$(1 - \delta_k)\|x\|_p \le \|C\|_p \le (1+\delta_k)\|x\|_p, \tag{5}$$

where δ_k is an isometry constant of matrix C, which must not be too far from zero, and p is equal to 1 (or very close) [9]. In this paper, unlike the compressive sensing approach where random projection is prevented when packet loss occurs in the network, we want approximate recovery of the data rather than no recovery.

2.2 RLNC Recovery

On the client side, l linearly independent symbols are received, and recovering the source data is attempted.

$$\begin{bmatrix} y(1) \\ \vdots \\ y(l) \end{bmatrix} = [c_1 \cdots c_n] * \begin{bmatrix} x(1) \\ \vdots \\ x(n) \end{bmatrix}. \tag{6}$$

Here, c_n is the column vectors of $C^{l\times n}$ network transform matrix. If the rank of network transform matrix C is equal to N (i.e., $l = N$ full rank), then x is uniquely determined as

$$\hat{x} = C^{-1} y, \tag{7}$$

where C^{-1} is the inverse of the network transform matrix C, which can be obtained by a common approach, such as Gaussian elimination over a finite field. However, if the size of total received symbols is less than original signal, $l < N$ (i.e., rank deficient), then x has no unique solution and may have infinite solution. Thus, in order to solve the rank deficiency problem, we rewrite the received rank deficient matrix (6) as follows:

$$\begin{bmatrix} y(1) \\ \vdots \\ y(m) \end{bmatrix} = \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{bmatrix} * \begin{bmatrix} x(1) \\ \vdots \\ x(n) \end{bmatrix}, \tag{8}$$

where y^m is the random projection of x^n and $C^{m\times n}$. Our network transfer matrix fulfills the RIP(p) condition, and the

Algorithm 1 Recovery Algorithm.

Initialization: received symbol y, network transfer matrix C,
if $rank(C) = N$ **then**
  $\hat{x} = C^{-1}y.$
else
  $\hat{x} = arg\min \|x\|_0 \quad subject \quad to \quad Cx = y.$ //Sparse recovery
end if

source data of x are k-sparse in Ψ domain. Thus, we redefine the rank deficiency problem into compressive sensing recovery theory. e.g., if $rank(C) = l$ and $l < n$, we solve this problem using a sparse recovery algorithm. After recovering the x value with the recovery algorithm, the value s of the original data can be obtained by (1). An example of recovery algorithm is given in Algorithm 1.

2.3 Sparse Recovery Approach

For the sparse recovery algorithm, we redefine the rank deficiency problem. The signal x represents a vector of length n with a k-sparse signal over the finite field. And y is error occurred data with the vector length is equal to m ($m \ll n$), that linear combination by x and $C^{m\times n}$ matrix over the finite field. To recover x, we must solve the minimization problem, defined as

$$\hat{x} = arg\min \|x\|_0 \quad subject \quad to \quad Cx = y, \tag{9}$$

where $x \in \mathbb{F}^n$ and $C \in \mathbb{F}^{m\times n}$.

In standard compressive sensing theory, operations are performed over $\mathbb{R}$. In addition, operations are processed in real field, and this minimization problem is non-convex and considered NP-hard. Thus, a standard relaxation approach used in compressive sensing is to replace l_0 norm by l_1 norm. There are many algorithms in the real field to solve this optimization problem (i.e., lasso, basis pursuit, and orthogonal matching pursuit); however, finite alphabet optimization problems are more complex, comparing with their real-valued parts. For example, although it is difficult to find the logarithm inversion over a finite field, the same is easy with real values. Some research into solving the optimization problem has been conducted in the finite field. The first paper of compressive sensing over the finite field introduces the theoretical result of error exponent outcome [10]. The probability of existence of a signal sparser than the input signal that matches the measurements using random finite field sensing matrices was calculated. A greedy finite field algorithm that works with a sparse binary sensing matrix over the finite field has been developed [11]. They showed a comparison between real and finite field CS recovery performance for both generated data and sparse gray scale images. Previous studies are primarily theoretical and were performed in an ideal environment to determine the possibilities offered by compressive sensing over finite fields with sparse sensing matrices. We employ the ideal l_0 norm (equal to the Hamming distance) to solve (9), which is a well-known greedy algorithm of orthogonal matching pursuit (OMP) [12].

At step $i \geq 1$, the optimal index J is selected. The classic OMP over real field selects columns that minimizing the distance $J = arg\min_j \left\|\alpha c_j - r^i\right\|_2$ where $\alpha = r^i c_j / \left\|c_j\right\|_2$ [12]. Nevertheless, l_2 norm is undefined over a finite field, where Hamming distance can be used. The distance between columns and residual is calculated as $J = arg\min_j \left\|\alpha c_j - r^i\right\|_0$, where α is the minimum value that J when $\alpha c_j - r^i = 0$. Thus, α is every feasible value of $r^i/unique(C)$ over the finite field, where $unique(C)$ is non-redundant value of matrices C. After find the nearest columns of C to r^i, $\hat{x} = C_J^+ y$ is calculated to estimate x. Here C_J^+ is not square matrix, and we use the pseudo-inverse. The full algorithm of the finite field OMP is given in Algorithm 2.

Algorithm 2 Finite Field OMP.

Initialization: $r^{(0)} = y$
while $(i \leq k)$ **do**
$J = arg\min_j \left\|\alpha c_j - r^i\right\|_0$
$\hat{x} = C_J^+ y$
$r^{i+1} = y - C_J \hat{x}$
$i = i + 1$
end while

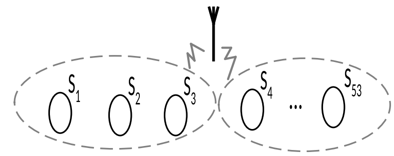

Fig. 1 Example of sensor network

3. Evaluation and Simulation

Here, we give an example of a sensor network that captures a source signal from another location. In this example, sensors are connected via wireless links, and the sensors transmit RLNC encoded data. Furthermore, sensors combine data using RLNC, wherein each sensor measures its own data and receives data from neighboring sensors. In this example, there are 53 sensors that measure the amplitude of a seismic signal at another location. An example of the distributed sensors is shown in Fig. 1. We assume that the neighboring sensor set is constructed of three sensors based on their location. Here, s_1, s_2, and s_3 are the measured amplitude of seismic data from sensors 1, 2, and 3, respectively. The measured signals are time-shifted and energy scaled by Gaussian distribution (i.e., $s_1 - s_2\ N(8, 0)$). The correlation of the signals increases as sensors get close to the neighboring sensors. Figure 2 shows the seismic signal measured by sensor 1. Each sensor captures a signal that processes a series of sampled values in a time window of size w (window size $w = 32$), and the size of finite field is 2^{10} (i.e., 2^p). Figure 3 shows the decoded data processed by the proposed algorithm, where packet loss ε, where 30% of total packet. The sparsity factor of network transfer matrix is 0.2 and the DCT transform matrix is used to select k-sparse signal. Since the k-sparse signal selected from DCT domain

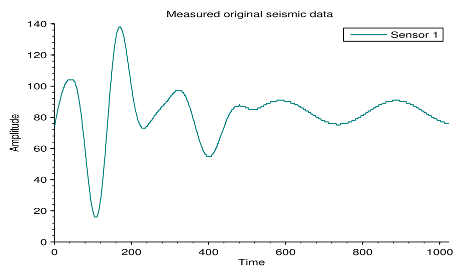

Fig. 2 Measured original seismic signal

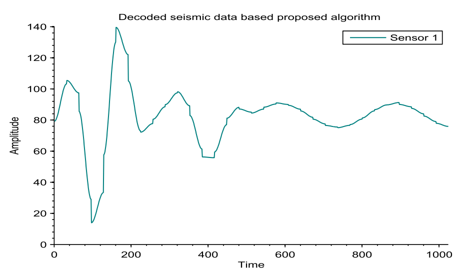

Fig. 3 Decoded seismic signal based on the proposed algorithm

Table 1 Average MSE for finite field OMP and classic OMP

ε	Finite Field OMP	Classic OMP
10%	0.0	4.558e-28
20%	0.0	5.366e-28
30%	0.0	6.999e-28

Table 2 Simulation result in NMSE (ε = 30%).

	Sensor 1	Sensor 2	Sensor 3
Approximate decoding	0.846	0.230	0.216
Proposed approach	0.010	0.005	0.015

is based on the assumption of high correlation between the neighboring sensors; thus, the recovered data of sensor 1 (Figs. 2 and 3) are very similar. However, there is a discontinuity in the recovered signal because of the k-sparse approximation process, i.e., approximation error. This discontinuity can be avoided if we select an appropriate k-sparse approximation method or the signal is sufficiently sparse in some domain.

The performance of our finite field OMP algorithm and the classic OMP algorithm [12] is compared over a real field when applied to the measured data from sensor 1, sensor 2, and sensor 3, where $w = 32$. The sensor's total length in the same period is n (i.e., $n = 3 \times w$) and has a sparsity k/n of approximately 14%. The measurement matrices of $C^{m \times n}$ are sparse matrix, where $m = n - \varepsilon$ and ε is set to 10%, 20%, and 30% of total signal n. Table 1 shows the average mean square error (MSE) of the reconstructed and original signal. The results confirm that the finite field OMP can recover the signal at a similar performance level compared to the classic OMP over a real field.

We compare the proposed method to an approximate decoding approach [2] because the approximate decoding approach uses similarity data of the sources to solve an underdetermined system. For fair comparison, we set the finite field size to 2^5, because in [2], this setting yields the best results when the source data are similar. Table 2 shows the normalized the mean square error (NMSE) of the proposed algorithm and the approximate decoding approach. The proposed algorithm outperforms the previous method. Therefore, we conclude that the proposed method is an effective solution for solving the all-or-nothing problem.

4. Conclusion

We have proposed a new framework for sparse recovery over a finite field with sparse random network transfer matrices that can overcome the all-or-nothing problem in NC. The sparse network transfer matrices are used for random linear NC and satisfy the RIP(p) property of the compressive sensing recovery condition. Simulation results demonstrate that the proposed approach can achieve better performance than previous methods. In the future, a theoretical analysis of the sparse network transfer matrices will be performed.

References

[1] R. Ahlswede, N. Cai, S.-Y.R. Li, and R.W. Yeung, "Network information flow," IEEE Trans. Inf. Theory, vol.46, no.4, pp.1204–1216, July 2000.
[2] H. Park, N. Thomos, and P. Frossard, "Approximate decoding approaches for network coded correlated data," Signal Processing (Elsevier), vol.93, no.1, pp.109–213, Jan. 2013.
[3] D. Silva, F.R. Kschischang, and R. Koetter, "A rank-metric approach to error control in random network coding," IEEE Trans. Inf. Theory, vol.54, no.9, pp.3951–3967, Sept. 2008.
[4] D.L. Donoho, "Compressed sensing," IEEE Trans. Inf. Theory, vol.52, no.4, pp.1289–1306, April 2006.
[5] S. Katti, S. Shintre, S. Jaggi, and D.K.M. Medard, "Real network codes," Proc. Forty-Fifth Annual Allerton Conference, 2007.
[6] S. Feizi and M. Medard, "A power efficient sensing/communication scheme: Joint source-channel-network coding by using compressive sensing," Proc. 49th Annual Allerton Conference on Communication, Control, and Computing, pp.1048–1054, 2011.
[7] M. Wang and B. Li, "How Practical is Network Coding?," IEEE International Workshop on Quality of Service 2006, Yale Univesrity, New Haven, Connecticut, pp.274–278, June 19-21, 2006.
[8] C. Cooper, "On the distribution of rank of a random matrix over a finite field," Random Structures and Algorithms, vol.17, no.3-4, pp.197–212, 2000.
[9] R. Berinde, A.C. Gilbert, P. Indyk, H. Karloff, and M.J. Strauss, "Combining geometry and combinatorics: a unified approach to sparse signal recovery," 2008 IEEE 46th Annual Allerton Conference on Communication, Control, and Computing, pp.798–805, 2008.
[10] S.C. Draper and S. Malekpour, "Compressed sensing over finite fields," in ISIT, pp.669–673, 2009.
[11] V. Bioglio, G. Coluccia, and E. Magli, "Sparse image recovery using compressed sensing over finite alphabets," 2014 IEEE International Conference on Image Processing (ICIP), pp.1287–1291, 2014.
[12] J.A. Tropp, and A.C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," IEEE Trans. Inf. Theory, vol.53, no.12 pp.4655–4666, Dec. 2007.