Joint Channel-Network Coding (JCNC) for Distributed Storage in Wireless Network*

Ning Wang and Jiaru Lin

Key Laboratory of Information Processing and Intelligent Technology,
Beijing University of Posts and Telecommunications
`bjxt713@gmail.com, jrlin@bupt.edu.cn`

Abstract. We propose to construct a joint channel-network coding (knosswn as Random Linear Coding) scheme based on improved turbo codes for the distributed storage in wireless communication network with k data nodes, s storage nodes (k<s) and a data collector. This framework extends the classical distributed storage with erasure channel to AWGN and fading channel scenario. We investigate the throughput performance of the Joint Channel-Network Coding (JCNC) system benefits from network coding, compared with that of system without network coding based only on store and forward (S-F) approach. Another helpful parameter: node degree (L) indicates how many storage nodes one data packet should fall onto. L characterizes the en/decoding complexity of the system. Moreover, this proposed framework can be extended to ad-hoc and sensor network easily.

Keywords: Joint Channel-Network Coding(JCNC), Distributed Storage, Wireless networks.

1 Introduction

Network coding [1] is a recent field in information theory that breaks with this assumption: instead of simply forwarding data, nodes may recombine several input packets into one output packets. This coding method can cope with the condition that source nodes disappear or inactive, which is the biggest trouble in the distributed communication. In this essay, we bring forward the scheme of distributed storage, where data packets generated by distributed sources are channel-coded first and sent to storage nodes through AWGN or Rayleigh channel, and then combined through network coding and saved at the storage nodes. System maintenance requirements and throughput can be significantly improved while adopting the integrity of stored information.

One of the key processes in distributed network storage is how to reconstruct the k data packets from any k out of s storage nodes, which is essentially the erasure channel coding problem [2]. Many codes used in this scenario have been investigated

---

* This work has been supported by the Doctor Fund of Beijing University of Posts and Telecommunications.

J. Zhou (Ed.): Complex 2009, Part I, LNICST 4, pp. 291 – 301, 2009.

on erasure channels. Erasure codes have been introduced in distributed storage in the OceanStore project [3]. In [4] the authors provide a random linear coding with a centralized server for distributed storage. Recently, fountain codes, like LT-code [5] and Raptor code [6], have been researched in distributed communication. All the codes introduced above work well for distributed storage in erasure channel. However, the coding structure introduced in this work can be used in wireless broadcast channel, like Gaussian or fading channel. We approximate a digital fountain basing turbo coding [7] [8] to broadcast source data and adopt random network coding to collect the data. Simulation results show that system with joint channel-network coding (JCNC) outperforms the system with only channel or erasure coding. It is the redundancy contained in the transmission of the relay on the storage nodes with JCNC that brings the throughput improvement.

The main innovations and contributions of this paper: we approximate a digital fountain based on turbo code, so the designed distributed storage system can be applied in Gaussian or fading channel instead of only in traditional erasure channel. Digital fountain breaks the tradition that the collector always receives a sequent stream of data packets, instead, the data source can product limitless encoding packets in digital fountain pattern. The simulation results show that the throughput of the system is greatly enhanced. It is network coding that brings the gratifying improvement. In addition, the optimal query intervals for the collector with different simulation conditions are provided in this paper, which reduces the total communication overhead greatly.

This paper is organized as follows: Related work about distributed network storage is discussed in section 2. We present the theory background, system model and assumption in section 3. Section 4 provides the simulation results and performance analysis. Finally, in section V we draw the conclusion of this work and discuss the future work.

2 Background and Related Work

2.1 Distributed Storage System

Distributed storage covers many key technologies about communication, e.g. file distribution, data replication and collection, distributed transaction mechanism, distributed timing and coordination, network security. Nowadays, the research about distributed network storage in the internet-based application and services is relatively interesting and mature. However, distributed media across wide areas and distributed network storage in wireless environment, e.g. sensor and ad-hoc network, pose an interesting challenge for present distributed technologies.

It is may be pretty to view distributed architecture as an idea, not just a technology. The book [9] gives a number of complex tradeoffs to be considered when managing the delivery, storage and collection of distributed data, which can not easily be duplicated at different sites.

2.2 Reed-Solomon (RS) Code

As a typical erasure code, RS code is one of Maximum Distance Separate (MDS) codes. The distinct advantage for RS code over simply replicating codes is that any d out of the s encoded fragments (storage nodes) suffices to recover the original d data fragments. In practice, however, decoding of RS code becomes expensive even infeasible in the network with large-scale nodes, just because the inverse matrix of generation matrix is required for the encoding. Standard algorithms for decoding RS code require exponential time. Maybe RS code is suitable on the condition that small blocks of data need to be encoded. Hard decoding is practically a fatal limitation for RS code applying to distributed network storage where the network scale is usually large or dynamic.

2.3 Low Density Parity Check (LDPC) Code

Low Density Parity Check (LDPC) LDPC (Low Density Parity Check) was proposed as an alternative with random construction [10], a random matrix with large weight. LDPC code relies on a parity check matrix based on bipartite graphs with the data nodes on the left and storage nodes on the right. Linear equations are produced between data blocks and coding blocks. The encoding and decoding is extremely fast [11]. However, since the LDPC codes are not MDS codes, coding blocks are required to reconstruct the original blocks. The overhead of application to distributed storage is too large. So LDPC structure needs to be modified for information distributed applications [10] [12].

2.4 Fountain Code

Fountain code was put forward by M.Luby in 1998. It was not until 2002 that was the feasible coding method raised. LT codes [5] and Raptor codes [6] are classical fountain codes in practice. The encoding of LT-code is XOR operation on d information symbols selected randomly according to the predetermined degree distribution of encoding symbol. For Raptor code, the M original blocks are preencoded into M' symbols firstly, and then only a successive function of M' suffice to recover the original information.

Compared to the limitation of slow encoding and decoding over large block sizes, fountain code breaks the source data into small blocks of packets and encodes over these blocks. The main idea of fountain code is that the encoding symbols can be transmitted limitlessly. An additional encoding block will be retransmitted when the receiver doesn’t receive the given symbol correctly, which eliminates the need for retransmission dramatically. The key feature of fountain code is that the receiver can reconstruct the original data from any d out of s encoding blocks (the original data is break into d blocks and encoded into s blocks). From above discussion, fountain code can be well applied in distributed network environment, e.g. internet, satellite networks and wireless sensor or ad-hoc networks.

2.5 Network Coding

The capacity of information stream from end to end in communication networks is determinate by the mini-cut of network digraph. However, traditional Store-Forward

(SF) scheme cannot reach the up-bound of maxflow-mincut theorem raised by Shannon. In 2000, Ahlswede.R and Li.S-Y.R brought forward the idea of network coding: routers can combine different information flows via encoding instead of only storing, thus available network resource can be maximum utilized.

The key feature of network coding is that the receiver encodes what it has received and then forwards the coding packets. There are two main benefits of this approach: potential throughput improvements and a high degree of robustness. For the server, the original data (d blocks) can be decoded as long as the rank of decoding matrix reaches d. The network decoding will be faster when the data nodes get more dispersed, so network coding is being paid much attention and applied to distributed network communication. Fast en/decoding and improvement on throughput promote the application and research of network coding recently, e.g. multicast capacity, cooperation communication, mesh network and sensor network.

3 System Model and Realization

3.1 System Model and Assumption

There are k data nodes, s (s>k) storage nodes with limited memory and one collector in our system shown in Figure.1. We assume one data node generates only one data packet and the data packets are sent to storage nodes independently. One data packet is sent to L storage nodes (selected randomly) uniformly at one time. We emphasize the parameter L and simulate the measurable value with different d (data nodes) and s (storage nodes). L, so-called data node degree, represents the number of edges in Figure.1, which influences the sparsity of the encoding matrix S and decoding complexity directly.

Before sent to storage nodes through AWGN or Rayleigh channel, channel coding is operated on the source data. In this work, we approximate a digital fountain based on turbo code. Thus the encoding message can be sent to the storage nodes limitlessly until the collector retrieves the original data.

The storage nodes play the role of channel decoding, re-encoding (network coding on storage nodes) and saving devices. We assume one storage node has the same limited memory as one data node. The storage nodes decode the message received from data nodes firstly. Only if decoded correctly, the message will be saved. When more than one packet is decoded correctly, they will be compressed into one packet via network coding.

The data collector queries the storage nodes at an interval and reconstructs the original data. The collector is assumed to have enough memory and power for decoding the k original data packets. We define the system throughput: k/n, where n is the number of storage nodes the collector needs to query for reconstructing the original k data blocks. It is assumed that the collector can retrieve the data from any k storage nodes, which is the critical feature and design criteria for distributed storage system, from the moment that k/n reaches 1. So many important parameters in our

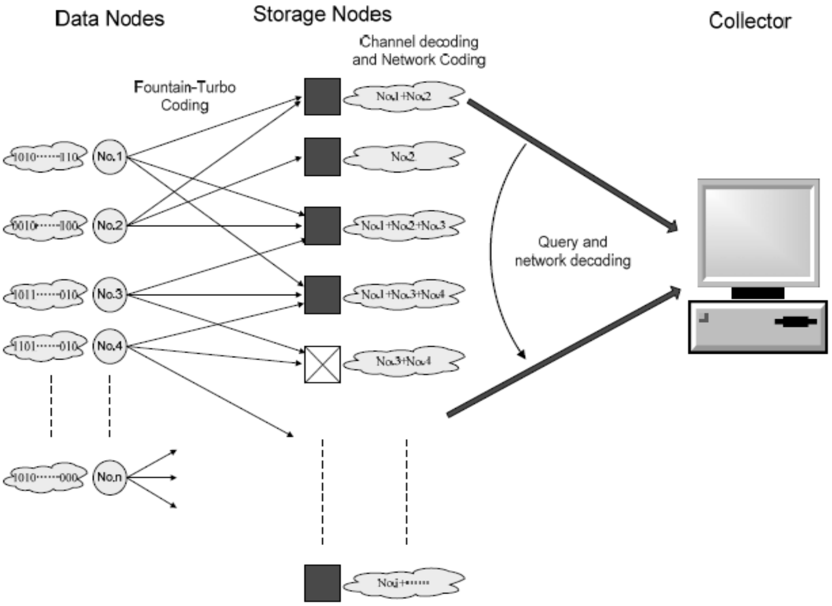

Fig. 1. System model: d data nodes, s storage nodes and 1 collector which can be set anywhere of the network

system are simulated at the condition k/n=1. We calculate the value of k/n whenever the rank of decoding matrix G' gets full. Network decoding at one SNR will be operated until k/n reaches 1. Also, the whole times of decoding matrix G' getting full before k/n reaches 1 is defined as the system communication time or source data reconstruction time (expressed as Time in section 4), representing the communication efficiency of the system.

3.2 System Realization

The main task in distributed storage system is that the k source data packets are saved in a redundant way in the s storage nodes and the collector can retrieve the original data by querying any k storage nodes in its vicinity. The channel encoding data packets fall into storage nodes randomly and random network coding (will be detailed in next section) is operated on the storage nodes.

In this wireless distributed network storage system, turbo coding with good BER performance is operated on the source data through AWGN or fading channel. On the storage nodes, the channel coding packets will be channel-decoded firstly. If more than one packet has been decoded correctly by the storage nodes, the packets will be network encoded, not simple saving and forwarding (SF).

We assume that information is transmitted as vectors of bits which are of length u, represented as elements in the finite field F (2u). In this paper we consider random linear coding for transmission and compression of information in general multi-source multicast networks. The linear combination of packets is random network coding over a finite field F (2u). One Storage node may receive several packets from data nodes. The coefficients of the polynomial generated at one storage node are selected from 0 to 2u -1 randomly.

Any storage node S can be illustrated as Figure.2 [13]. For a linear approach, S j on a link j is a linear combination of processes Xi generated at node v = tail (j) and signals Di on incoming links.

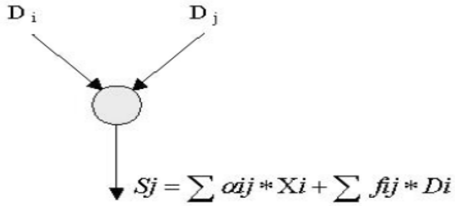

Fig. 2. Illustration of linear coding at a storage node

The decoding can be presented as:

$$Zi = \sum \eta ij * Si \tag{1}$$

As long as d (the number of source nodes) linear combinations are received, the source signal will be retrieved theoretically. Based on the random linear coding/decoding theory, this paper provides JCNC approach for wireless distributed scenario and analyses its performance.

Correspondingly, the row vectors of generation matrix G_{k*n} are random and independent absolutely and the data vector is D_{1*k}. Nonzero Element in storage encoding vector S_{1*n} represents the edge connecting data and storage nodes of the bipartite graph in Figure.1. This just shows the key property of our design: random and decentralized code structure.

$$S_{1*n} = \begin{bmatrix} d_1 & d_2 & \dots & d_k \end{bmatrix} * \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ . & . & . & . \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{bmatrix} \tag{2}$$

To retrieve the k data packets, it is clear that successful decoding requires a full rank matrix G'_{k*k} from G_{k*n}.The value of n will plus 1 when the collector queries data from one storage node to another. However, that of k plus 1 only when the rank of G' rises.

The realization process is described in figure 3,

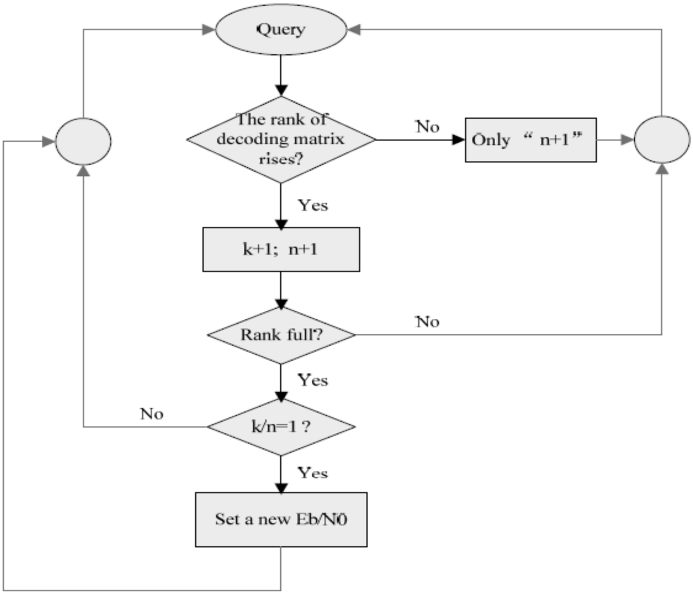

Fig. 3. Simulation flow chart: It shows the method of how to get the throughout of the System

4 Performance Evaluation

In this section, we provide the simulation results and analyze improvement performance of our scheme.

4.1 Throughput (k/n)

In this joint channel and network coding (JCNC) scheme, the performance of channel coding affects the whole capacity of system. So the average collector throughput (k/n) at several Eb/N0 is shown in Figure 4 and Figure 5, for AWGN channel and Rayleigh channel respectively.

Throughput of system with joint channel and network coding (JCNC) is compared with that of system without network coding (only fountain-channel coding). In the system with only channel coding (CC), every storage node saves only one packet. If one packet is decoded correctly and saved, the storage node will not accept new packet until next query. The count of n in only channel coding (CC) system is the same as JCNC system. However, the value of k plus 1 only when the channel decoding correctly (according to BER as shown in section 3.) and the number of packet is not repeated with that of former.

Distinctly from the simulation results, the throughput of JCNC is largely improved at the same SNR, which just benefits from the network coding. The throughput

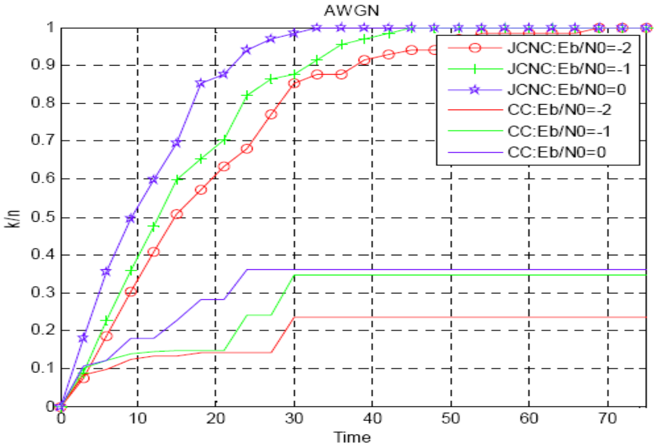

Fig. 4. Throughput of the system JCNC in AWGN Channel, compared with system CC. L=1, k/s=10%.

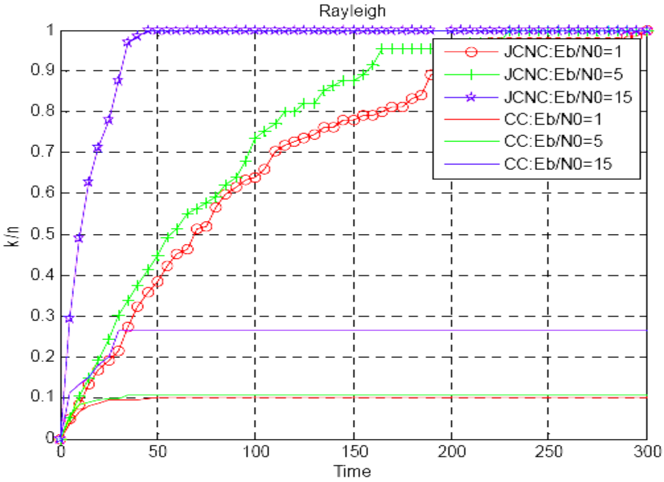

Fig. 5. Throughput of the system JCNC in Rayleigh Channel, compared with system CC. L=1, k/s=10%.

doesn't reach 0.5 even at high SNR (AWGN: Eb/N0= 0; Rayleigh: Eb/N0=15). Moreover, the throughput is related partially with the performance of BER as interpreted in section 3. And the BER is greatly improved as Eb/N0 rises as a result of channel coding. So it is clearly that the throughput gets better with the higher Eb/N0.

4.2 Node Degree (L)

As discussed in section 3, the cost and time delay of data query is minimal when k/n reaches 1. Moreover, some key parameters, the system overhead, query delay and en/decoding complexity, mainly relates with source node degree L. Emphasizing the effects of L in the system, we provide the simulation results at ideal channel condition (AWGN: Eb/N0=1dB, Rayleigh: Eb/N0=12dB).

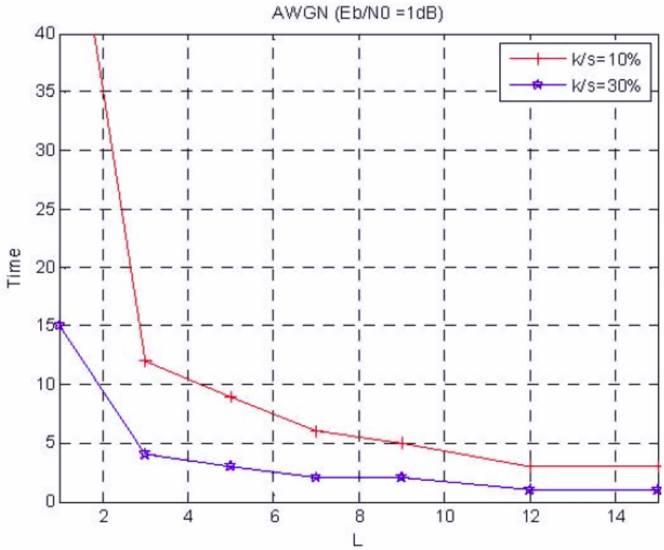

Fig. 6. Time of data reconstruction with different node degree (L) in AWGN Channel

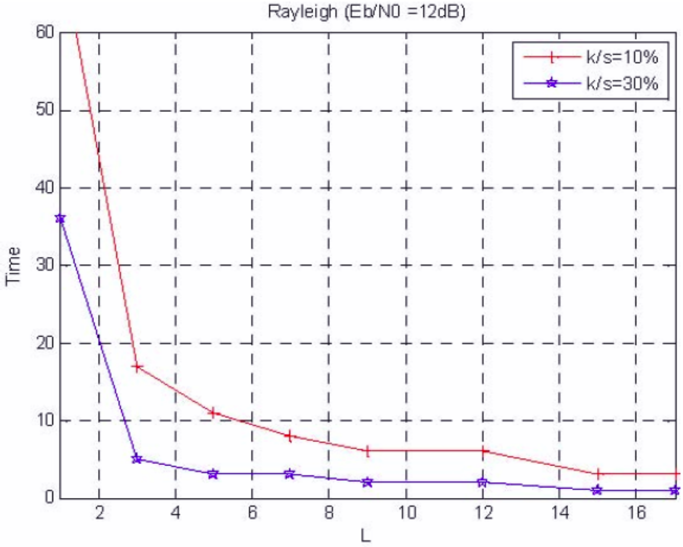

Fig. 7. Time of data reconstruction with different node degree (L) in Rayleigh Channel

Both Figure.6 and Figure.7 present the time of data reconstruction at the collector versus L with different network (k/s=10% and k/s=30%). Communication time decreases sharply with value of L rises from 1 to 4. However, the time of retrieving original data decreases much slowly, even remains constant, with L (L>5) rising. In addition, performance of Time in network with k/s=30% is better than network with k/s=10%.

As discussed in section 3, the cost and time delay of data query is minimal when k/n reaches 1. Moreover, some key parameters, the system overhead, query delay and en/decoding complexity, mainly relates with source node degree L. Emphasizing the effects of L in the system, we provide the simulation results at ideal channel condition. (AWGN: Eb/N0=1dB, Rayleigh: Eb/N0=12dB)

5 Conclusion and Future Work

We introduced the JCNC distributed storage in wireless network based on digital fountain turbo coding and network coding. The scheme extends the classical distributed storage with erasure channel to AWGN and fading channel. Simulation results confirm the improvement on system throughput. We provide another helpful parameter: source data node degree, which have guiding significance in practice. Finding effective algorithm to optimize d, s, L and time jointly and extending the system to ad-hoc or sensor network remain as future work.

References

1. Ahlswede, R., Cai, N., Li, S.-Y.R., Yeung, R.W.: Network Information Flow. IEEE-IT 1(46), 1204–1216 (2000)
2. Dimakis, A.G., Prabhakaran, V., Ramchandran, K.: Decentralized erasure codes for distributed networked storage. IEEE Transaction on Information Theory 52, 2809–2816 (2006)
3. Rhea, S., Eaton, P., Kubiatowicz, P. J.: The OceanStore prototype. In: Proc. USENIX File and Storage Technologies (FAST) (2003)
4. Acedanski, S., Deb, S., Médard, M., Koetter, R.: How Good is Random Linear Coding Based Distributed Networked Storage. In: NetCod 2005 (2005)
5. Luby, M.: LT codes. In: Proc. 43rd Annual IEEE Symposium on Foundations of Computer Science (2002)
6. Shokrollahi, A.: Raptor Codes, Technical Report DR2003-06-001 Digital Fountain (June 2003)
7. Jenkac, H., Hagenauer, J., Mayer, T.: The Turbo-Fountain and its Application to Reliable Wireless Broadcast. In: Information Theory Workshop (2006)
8. `http://www.ecse.rpi.edu/Homepages/shivkuma/teaching/sp2001/readings/digital-fountain`
9. Coulouris, J.D., Kindberg, T.: Distributed Systems: Concepts and Design, 4th edn. Addison Wesley/Pearson Education (2005)
10. Kameyama, H., Sato, Y.: Erasure Codes with Small Overhead Factor and Their Distributed Storage Applications. In: CISS 2007, 41st Annual Conference on Information Science and System (2007)

11. Plank, J.S., Thomason, M.G.: On the Practical use of LDPC Erasure Codes for Distributed Storage Applications. Technical Report CS-03-510, University of Tennessee (2003)
12. Plank, J.S., Thomason, M.G.: A Practical Analysis of Low-Density Parity-Check Erasure Codes for Wide-Area Storage Applications. The International Conference on Dependable Systems and Networks. IEEE, Los Alamitos (2004)
13. Ho, T., Medard, M., Shi, J., Effros, M., Karger, D.R.: On Randomized Network Coding, `http://web.mit.edu`