2011 International Symposium on Nonlinear Theory and its Applications
NOLTA2011, Kobe, Japan, September 4-7, 2011

Scale-free LT Codes and Its Performance on BEC Channels

Zhiliang ZHU†, Yuli ZHAO*, Francis C.M. LAU‡ and Hai YU†

†Faculty of Software College, Northeastern University, Shenyang, 110819, P. R. China
* College of Information Science and Engineering, Northeastern University, Shenyang, 110819, P. R. China
‡Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong
Emails: *susansimon986@163.com, ‡encmlau@polyu.edu.hk

Abstract— The degree distribution of the encoded symbols plays a crucial role in the design of a LT code — a type of channel codes. It affects the encoding and decoding cost/complexity as well as the error performance of the code. In this paper, we apply the shortest "average-path-length" property of scale-free networks to the construction of LT codes. In a "scale-free" LT code, the encoded symbols will follow a modified power-law degree distribution. We will also compare the characteristics and decoding performance of "scale-free" LT codes with traditional LT code constructed with robust soliton distributions.

1. Introduction

Reliable transmission of data over various channel environments has been the subject of much research. For the most part, reliability is realized by using some error control methods such as appropriate protocols and channel coding [1, 2]. Protocols relying on positive acknowledgement feedbacks [1], however, perform unsatisfactorily when packets are sent over heavily impaired channels. Channel coding can provide a stable error performance at the receiver for a time-invariant channel environment [2]. Yet, in the case of time-varying channel environments, decoding failures occur whenever the channel degradation exceeds the error-correction capability of the code.

Fountain codes [3], including LT codes [4] and Raptor codes [5], are one kind of rateless channel coding method that addresses the aforementioned issues. Given a set of K input (data) symbols, a LT code produces potentially an unlimited number of encoded symbols. Moreover, the degree of an encoded symbol, i.e., the number of input symbols that the encoded symbol is connected to, should follow a certain distribution in order to optimize the performance of the code. The receiver, based on the encoded symbols received, tries to recover all the (input) data symbols by passing messages among the encoded symbols and the decoded symbols. By modifying the degree distribution of the encoded symbols of LT codes, improvements on the encoding/decoding complexity and error performance are made [6, 7].

In recent years, properties of complex networks have been successfully applied to solve engineering problems such as traffic prediction [8], blackout prediction [9], construction of low-density parity-check codes [10] as well as modeling of call networks [11]. A complex network consists of nodes and connections. Properties of various kinds of complex networks including random networks, regular lattices, small-world networks and scale-free networks have been extensively studied [12, 13, 14]. It has also been proven that for the same number of connections, scale-free networks possess the shortest average-path-length (APL) among the aforementioned complex networks.

In this paper, inspired by the shortest APL property of scale-free networks, we propose constructing LT codes with a modified power-law encoded-symbol degree distribution. We called LT codes with such a property *scale-free LT codes* (SF-LT code).

2. LT Codes

LT codes were invented by Luby and were applied to binary erasure channel (BEC) environments [4]. For all types of erasure channels, the original data symbols can be recovered after a sufficient number LT-encoded symbols have been received. Here, a symbol can represent one bit or a sequence of bits [3, 4].

Assume that there are K (input) data symbols denoted by $s_1, s_2, \ldots, s_{K-1}, s_K$. In the encoding process, each encoded symbol c_n ($n = 1, 2, \ldots$) is assigned a degree d_n (the number of connections to the input symbols) ranging from 1 to K according to a given degree distribution. Moreover, the degree distribution of the encoded symbols should meet the following principles [4] — a few encoded symbols should have high degrees in order to ensure that no input symbols are left connected; and most encoded symbols should have low degrees so that the decoding process (to be discussed below) can get started and keep going. Then d_n distinct input symbols are randomly selected as neighbors of the encoded symbols. Afterwards, the value of the encoded symbol is derived by XOR-ing the d_n distinct input symbols[1]. The encoding process eventually defines a bipartite (Tanner) graph [15] connecting the K input symbols and the encoded symbols. Figure 1 shows one example with 10 input symbols.

Assuming a BEC channel, the receiver keeps receiving encoded symbols though some encoded symbols are lost in

[1]In case each input symbol contains more than one bit, the XOR is performed in a bit-wise manner.

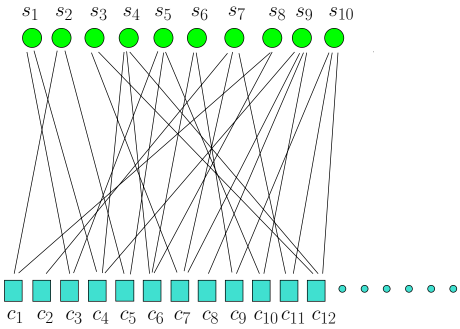

Figure 1: Tanner graph of a LT code.

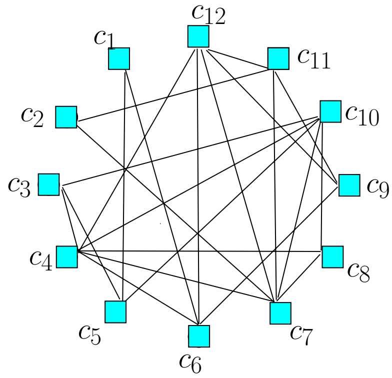

Figure 2: Complex network formed based on the encoded symbols of the LT code in Fig. 1

the transmissions. Decoding of the LT code commences as soon as a sufficient number of symbols are received. The iterative decoding algorithm is described as follows.

1. Find an encoded symbol c_j of degree 1[2] and denote its unique neighbor (input data symbol) as s_i.

2. Copy the value of c_j to s_i, i.e., assign $s_i = c_j$.

3. Suppose s_i is connected to l other encoded symbols in addition to c_j. Denote $c_{j,1}, c_{j,2}, \ldots, c_{j,l}$ as the other encoded symbols connected to s_i. Update $c_{j,k}$ using $c_{j,k} = c_{j,k} + s_i$, for $k = 1, 2, \ldots, l$.

4. Remove input node s_i and all edges emanating from s_i in the Tanner graph. If all the K input symbols have been recovered, stop the decoding process.

5. Go back to Step 1.

Note that a decoding failure is declared if the input symbols are not fully recovered when no more encoded symbols are received.

An ideal decoding behavior can be accomplished in theory when the encoded-symbol degree follows the ideal soliton distribution given by [3, 4]

$$\rho(d) = \begin{cases} \frac{1}{K} & \text{for } d = 1 \\ \frac{1}{d(d-1)} & \text{for } d = 2, 3, \ldots, K. \end{cases} \quad (1)$$

However, the ideal soliton distribution works poorly in practice, as at some point in the decoding process there would be no degree-1 encoded symbols. As a consequence, the whole decoding process fails.

The robust soliton distribution has solved this problem by introducing two parameters c and δ. Here, $c \in (0, 1)$ is a free parameter and δ represents the maximum failure probability of the decoder when $N = K + O(\sqrt{K}\ln^2(K/\delta))$ encoded symbols are received [4]. Using the robust soliton distribution ensures that the expected number of encoded symbol with degree 1 at each iteration is large. Consequently, the decoding process can continue and hence the decoding can succeed with a high probability. Finally, the robust soliton distribution $\mu(d)$ is defined as [4]

$$\mu(d) = \frac{\rho(d) + \tau(d)}{\sum_{d=1}^{K}(\rho(d) + \tau(d))} \qquad \forall\, d = 1, 2, \ldots, K-1, K \quad (2)$$

where

$$\tau(d) = \begin{cases} \frac{S}{K}\frac{1}{d} & \text{for } d = 1, 2, \ldots, \lfloor(K/S)\rfloor - 1 \\ \frac{S}{K}\ln(S/\delta) & \text{for } d = \lfloor(K/S)\rfloor \\ 0 & \text{for } d > \lfloor(K/S)\rfloor. \end{cases} \quad (3)$$

3. Proposed Scale-free LT Codes

In the previous section, we have shown that the decoding process of the LT code over a BEC channel is based on passing "messages" iteratively between the encoded symbols and the input symbols. Since the "messages" originate from the encoded symbols, we can visualize the message-passing process by forming a complex network consisting of the encoded symbols. Representing an encoded symbol by a node in a complex network, we connect two nodes when the two corresponding encoded symbols share a common input symbol (i.e., the two encoded symbols are connected to the same input symbol). In Fig. 2, we illustrate the complex network formed based on the encoded symbols of the LT code in Fig. 1.

When $N \geq K$ encoded symbols have been received and with a sufficient number of degree-1 encoded symbols, "messages" from the degree-1 encoded symbols are first passed to the neighboring nodes in the corresponding complex network. As a messages is passed, the node passing the message and its connections can be removed from the network. This message-passing algorithm continues as long as there are remaining nodes with degree 1. When all the nodes and hence the connections are removed, the input symbols have been decoded successfully (assuming that each input symbol is connected to at least one of the encoded symbols received). Otherwise, some input sym-

[2] If no degree-1 encoded symbol exists, wait until a degree-1 encoded symbol is received from the channel.

bols are yet to be decoded and more encoded symbols need to be collected.

Referring to Fig. 2, the path length between two nodes represents the number of iterations required for the "message" originated from one node to be passed to other node. Thus, for a given network size (number of encoded symbols) N, there can be a higher chance of successful decoding if the average path length among the nodes is shorter. It is well-known that among random networks, regular lattices, small-world networks and scale-free networks, scale-free networks provide the shortest APL for a fixed number of nodes and a fixed number of connections [10]. Moreover, scale-free networks provide the smallest number of connections for a fixed number of nodes and a fixed APL.

With the aforementioned feature in mind, we propose constructing LT codes with the requirement that networks formed by the encoded symbols should be scale-free. We make use of the theorem in [16], which states that if the degree distribution of one set of nodes in a bipartite graph follows a power-law distribution, the degree distribution of the unipartite graph formed by this set of nodes should follow a power-law distribution with the same exponent. Consequently, all we need is to construct a LT code with its encoded symbols following a power-law degree distribution. Then, the complex network formed by the encoded symbols alone will also follow a power-law degree distribution with the same exponent.

For a pure scale-free distribution, there exist a large number of nodes with low degrees and a small number of nodes with large degrees. Here, we slightly modify the pure scale-free distribution by limiting the fraction of encoded symbols with degree 1. Such a restriction is quite common in many complex networks. As a result, we define a scale-free LT code as follows.

Definition 1: We define a scale-free LT (SF-LT) code as a LT code with its encoded-symbol degree following a modified power-law distribution. The distribution is further given by

$$\lambda(d) = \begin{cases} P_1 & \text{for } d = 1 \\ Ad^{-\gamma} & \text{for } d = 2, 3, \ldots, K-1, K \end{cases} \tag{4}$$

where $P_1 \in [0.1, 0.3]$ determines the fraction of encoded symbols with degree 1 (called the initial-ripple fraction); γ is the characteristic exponent; and A is the normalizing coefficient to ensure $\sum_{d=1}^{K} \lambda(d) = 1$.

4. Results and Discussions

We construct SF-LT codes for input symbol length $K = 1000$ based on the degree distribution in (4). We use three different sets of parameters — (i) $P_1 = 0.1$ and $\gamma = 1.9$; (ii) $P_1 = 0.1$ and $\gamma = 2.0$; (iii) $P_1 = 0.2$ and $\gamma = 1.9$ — for constructing the SF-LT codes. The average number of degrees for each encoded symbol, denoted by $\bar{d}$, is calculated by substituting (4) into $\bar{d} = \sum_{d=1}^{K} d\lambda(d)$ and is listed in Table 1. Moreover, for every input-symbol set, we generate the corresponding SF-LT codes with $2K = 2000$ encoded symbols based on the aforementioned parameter sets. This is repeated 1000 times for every set of parameters. Then, we measure the average number of XOR operations $\bar{x}$ required to encode each encoded symbol. In addition, the encoded symbols are sent sequentially to the receiver (with no erasures) until the receiver can recover the input symbols correctly. We then record the average number of XOR operations Ψ used to recover the input symbols. The results in Table 1 show that among the three sets of parameters, the one using $P_1 = 0.1$ and $\gamma = 2.0$ to construct SF-LT codes possesses the least complexity in terms of average number of XOR operations to encoded one symbol and average number of XOR operations to recover the input symbols.

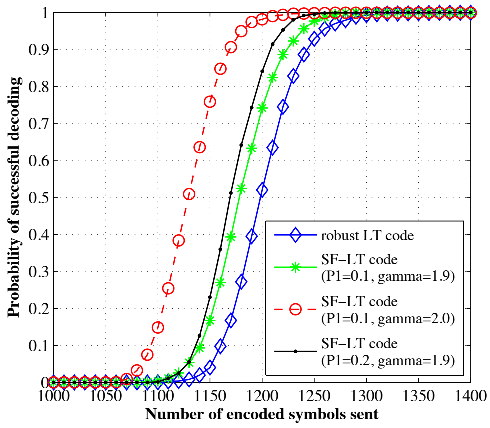

Figure 3: Probability of successful decoding versus the number of encoded symbols sent for robust LT code and SF-LT codes ($P_1 = 0.1$ and $\gamma = 1.9$; $P_1 = 0.1$ and $\gamma = 2.0$; $P_1 = 0.2$ and $\gamma = 1.9$). $K = 1000$.

We also construct LT codes based on the robust soliton degree distribution given in (2) for comparison purpose. We arbitrarily set $\delta = 0.02$ and evaluate the performance of the code for different values of c while K is fixed at 1000. We find that using $c = 0.12$ gives the best overall probability of successful decoding. (Details not shown due to space limitation.) Thus, we use the parameters $\delta = 0.02$ and $c = 0.12$ in constructing robust LT codes. In Table 1, we show the characteristics of the robust LT codes constructed.

Next, we compare the performance of the our proposed SF-LT codes with the robust LT code. 2000 different sets of encoded symbols are formed according to the corresponding encoded-symbol degree distribution. These encoded symbols are subsequently sent to the receiver. Figure 3 shows the results for the perfect channel, i.e., no symbols are erased. The results indicate that the SF-LT codes always give a higher probability of successful decoding for a given number of encoded symbols sent. Further, it is clearly seen that the SF-LT code with $P_1 = 0.1$ and $\gamma = 2.0$ significantly outperforms the robust LT code. The results

Table 1: Characteristics of the Codes Constructed.

Code	**Parameters**	**Average degree of an encoded symbol** $\bar{d}$ (K = 1000)	**Average no. of XOR operations to encode one symbol** $\bar{x}$ (K = 1000)	**Average no. of XOR operations to recover the input symbols** Ψ (K = 1000)
LT code (robust soliton)	$c = 0.12, \delta = 0.12$	9.07	8.023	9880
SF-LT code	$P_1 = 0.1, \gamma = 1.9$	11.67	10.546	12691
	$P_1 = 0.1, \gamma = 2.0$	9.26	8.204	9475
	$P_1 = 0.2, \gamma = 1.9$	10.392	9.376	11183

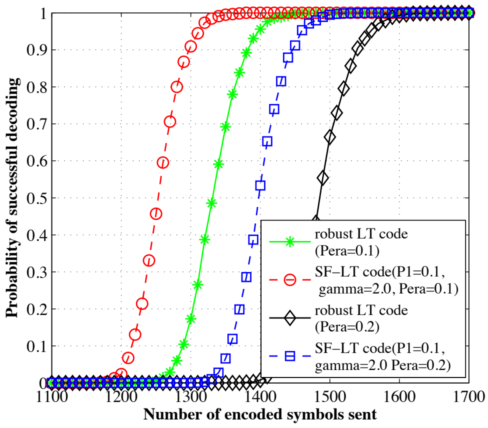

Figure 4: Probability of successful decoding versus the number of encoded symbols sent for robust LT code and SF-LT codes ($P_1 = 0.1$ and $\gamma = 2.0$). Erasure probability $P_{era} = 0.1$ and 0.2, $K = 1000$.

in Table 1 also indicate these two codes require similar encoding and decoding complexities.

Finally, we compare the performance of the SF-LT code (with $P_1 = 0.1$ and $\gamma = 2.0$) with that of the robust LT code over a BEC environment. Figure 4 shows that the SF-LT code always outperform the robust code when the erasure probability $P_{era} = 0.1$ and 0.2.

5. Conclusion

In this paper, we propose a novel type of LT code — scale-free LT (SF-LT) codes — in which the encoded symbols obeys a modified power-law degree distribution. The results show that the encoding and decoding complexities of SF-LT codes and LT codes are very similar. Yet, for a fixed number of encoded symbols sent, the SF-LT codes can provide better decoding performance compared with the robust LT codes under both ideal channel and BEC conditions.

Acknowledgments

The work described in this paper was supported by the National Natural Science Foundation of China (Grant No. 60872040), the Program for Liaoning Excellent Talents in University and the Natural Science Foundation of Liaoning Province, China (Grant No. 20082037), and the Fundamental Research Funds for the Central Universities (Grant No. N100604007).

References

[1] W. Stallings, *Data and Computer Communications*. Ninth edition, Prentice Hall, 2011.

[2] J. G. Proakis, *Digital communications*. Third edition, McGraw-Hill, Singapore, 1995.

[3] D. MacKay, "Fountain codes," *Communications, IEE Proceedings-*, vol. 152, no. 6, pp. 1062 – 1068, 2005.

[4] M. Luby, "Lt codes," in *Foundations of Computer Science, 2002. Proceedings. The 43rd Annual IEEE Symposium on*, 2002.

[5] A. Shokrollahi, "Raptor codes," *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2551 –2567, 2006.

[6] H. Zhu, G. Zhang, and G. Li, "A novel degree distribution algorithm of lt codes," in *Communication Technology, 2008. ICCT 2008. 11th IEEE International Conference on*, 2008, pp. 221 –224.

[7] Q. Zhou, L. Li, Z.-Q. Chen, and J.-X. Zhao, "Encoding and decoding of lt codes based on chaos," in *Innovative Computing Information and Control, 2008. ICICIC '08. 3rd International Conference on*, 2008, p. 451.

[8] Z. Gao and K. Li, "Evolution of Traffic Flow with Scale-Free Topology," *Chinese Physics Letters*, vol. 22, pp. 2711–2714, Oct. 2005.

[9] B. Gou, H. Zheng, W. Wu, and X. Yu, "Probability distribution of blackouts in complex power networks," in *Proc. IEEE Int. Symp. Circ. Syst.* New Orleans, USA, 2007, pp. 69–72.

[10] X. Zheng, F. C. M. Lau, C. K. Tse, Y. He, and S. F. Hau, "Application of Complex-Network Theories to the Design of Short-length LDPC Codes," *IET Communications*, vol. 3, no. 10, pp. 1569–1577, Oct. 2009.

[11] W. M. Tam, F. C. M. Lau, and C. K. Tse, "Complex-network modeling of a call network," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 56, no. 2, pp. 416–429, Feb. 2009.

[12] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[13] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, no. 6825, pp. 268–276, Mar. 2001. [Online]. Available: http://dx.doi.org/10.1038/35065725

[14] M. E. J. Newman, "The structure and function of complex networks," *SIAM Review*, vol. 45, no. 2, pp. pp. 167–256, 2003. [Online]. Available: http://www.jstor.org/stable/25054401

[15] X. Zheng, F. C. M. Lau, C. K. Tse, and S. C. Wong, "Study of Bifurcation Behavior of LDPC Decoders," *International Journal of Bifurcation and Chaos*, vol. 16, no. 11, pp. 3435–3449, 2006.

[16] J. L. Guillaume and M. Latapy, "A realistic model for complex networks," *Arxiv preprint cond-mat/0307095*, 2003.