2016 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2016, Yugawara, Japan, November 27th-30th, 2016

Robust Scale-free Luby Transform Code and Its Performance

Yuli Zhao† and Francis C.M. Lau‡, Zhiliang Zhu†, Hai Yu† and Li Guo#

†Software College, Northeastern University, Shenyang, China
#School of Computer Science and Engineering, Northeastern University, Shenyang, China
‡Department of Electronic & Information Engineering, Hong Kong Polytechnic University, Hong Kong;
and Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, China.
Email: zhaoyl@swc.neu.edu.cn, francis-cm.lau@polyu.edu.hk

Abstract—Compared with Luby Transform (LT) codes having an ideal/robust soliton degree distribution, LT codes with encoded-symbol degree following a modified power-law distribution (scale-free LT codes) have been shown to possess a higher probability of successful decoding and a lower encoding/decoding complexity when the information symbol length ranges from 512 to 2048. In an attempt to reduce the size of the initial ripple set of scale-free LT (SF-LT) codes so as to prevent the ripple set from becoming empty, a new class of LT codes, namely robust SF-LT (RSF-LT) codes, is proposed in this work. The performance and characteristics of the proposed RSF-LT code are compared with those of other LT codes including robust LT code, SF-LT code, and "LT code with decreasing ripple set". Results show that the proposed RSF-LT code outperforms the other codes with respect to average overhead, encoding/decoding efficiency and probability of successful decoding.

1. Introduction

Luby transform (LT) codes are the first type of practical rateless code [1]. It is originally designed for reliable data transmission over a binary erasure channel (BEC), which is suitable for the modeling of the Internet. Recently, LT codes have also been discussed for use in mobile multimedia broadcasting, wireless sensor networks, etc. [2, 3].

An LT code is capable of generating an unlimited number of encoded symbols based on a source message of length K. Regardless of the erasure probability of a BEC, an LT decoder can recover the original K input symbols when $(1+\alpha)\times K$ encoded symbols have been received. Here α is a real number slightly larger than 0. Consequently, LT code is a near-optimal channel code for all erasure channels.

Whether an LT code is well designed or not is determined by the degree distribution of its encoded symbols [4]. The ideal soliton distribution is the first degree distribution used to construct LT codes [1]. LT codes based on such a distribution can theoretically keep the ripple size always equaling one in the decoding process. Therefore, such a design avoids any redundancy and is optimal. However, any fluctuation around this expected behavior results in a lack of degree-1 encoded symbols and hence an unsuccessful decoding [5]. To deal with the above problem, a robust soliton distribution has further been proposed and it aims at maintaining a ripple size larger than one in the whole decoding process [1]. Results show that LT codes based on the robust soliton distribution outperform the original LT codes.

In [6], an LT code that can maintain the ripple size to a pre-defined constant during the decoding process has been proposed. In [7], LT codes with decreasing ripple size are designed and analyzed. The results indicate that such LT codes are capable of producing a higher performance. In [8], using the shortest-average-path-length property of scale-free networks, a class of scale-free LT (SF-LT) codes has been proposed. It has further been shown that SF-LT codes outperform LT codes based on robust soliton distribution and LT codes based on suboptimal distribution. In this paper, a new class of LT codes, namely robust SF-LT (RSF-LT) codes, is proposed and investigated.

2. Proposed Robust SF-LT code

In [8], a SF-LT code with the degree of the encoded symbols following a modified power-law distribution has been proposed. Specifically, the distribution is given by

$$\tau(d) = \begin{cases} P_1, & d = 1 \\ Ad^{-\gamma}, & d = 2, 3, ..., K-1, K. \end{cases} \tag{1}$$

where P_1 is the fraction of degree-1 encoded symbols; γ is the characteristic exponent; and A is a normalizing coefficient to ensure $\sum_{d=1}^{K} \tau(d) = 1$.

We defined a *released encoded symbol* as an encoded symbol whose degree becomes 1 during the iterative decoding process, and a *ripple set* as the set of input symbols which are connected to the released encoded symbols. Assume that at the end of each iteration in the decoding process, the neighboring input symbols of a newly released encoded symbol are not elements in the ripple set. Suppose $(1+\alpha)\times K$ encoded symbols have been received to recover K input symbols. Then, the theoretical evolution of

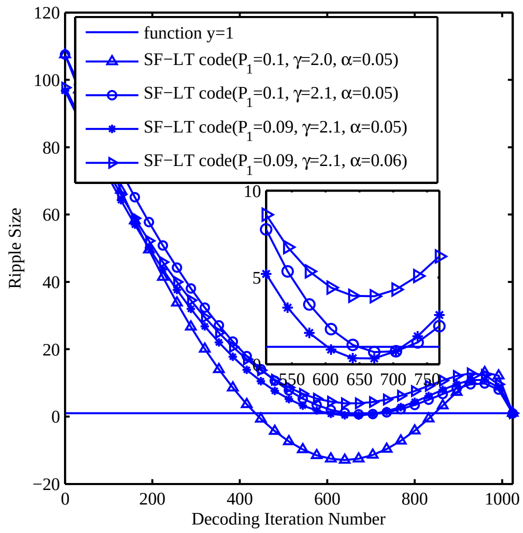

Figure 1: The theoretical ripple evolution of the SF-LT codes when $K = 1024$.

the ripple size can be calculated using [9]

$$\begin{cases} \Psi^K(i) = (1+\alpha)Kp(i) \qquad i = 1,2,\ldots,K \\ \Psi^{L-1}(1) = \Psi^L(1) - 1 + \frac{2(L-\Psi^L(1))}{L(L-1)}\Psi^L(2) \\ \Psi^{L-1}(i) = \Psi^L(i) - \frac{i}{L}\Psi^L(i) + \frac{i+1}{L}\Psi^L(i+1) \\ \qquad i = 2,3,\ldots,L-1 \\ \Psi^{L-1}(L) = 0 \end{cases} \tag{2}$$

where $\Psi^L(i)$ is the number of degree-i input symbols left in the decoding process when L input symbols remain unprocessed, and $p(i)$ denotes the probability of an encoded symbol having a degree i. Theoretically, the ripple size should be no less than 1 throughout the whole decoding process to prevent the decoding process stopped prematurely.

The theoretical evolution of the ripple size for a SF-LT code can be evaluated by substituting Eq.(1) into Eq.(2). Fig. 1 plots the ripple evolution of SF-LT codes when $K = 1024$. The results indicate that the SF-LT code can recover the original input symbols when the overhead factor is $\alpha = 0.06$ using the parameter set $P_1 = 0.09$ and $\gamma = 2.1$. However, when α is reduced to 0.05, the ripple size will become smaller than 1 at a certain point and the SF-LT codes will not be able to recover the input symbols.

In this paper, the characteristics of the ideal soliton distribution are applied to the design of SF-LT codes, forming the proposed robust SF-LT code. The aim is to decrease the probability that the ripple set becoming empty.

Definition 1 *A robust scale-free LT code, denoted as RSF-LT code for short, is defined as an LT code with the degree of the encoded symbols following a distribution given by*

$$\mu(d) = \frac{\rho(d) + \tau(d)}{\sum_{i=1}^{K}(\rho(i) + \tau(i))} \tag{3}$$

where $\rho(d)$ is the ideal soliton distribution expressed as

$$\rho(d) = \begin{cases} 1/K & d = 1, \\ \frac{1}{d(d-1)} & d = 2,3,...,K \end{cases} \tag{4}$$

and $\tau(d)$ is the modified power-law degree distribution given in Eq.(1).

The encoding and decoding complexity are both going to scale linearly with the number of edges in the Tanner graph. It has been shown in [5] that when the number of encoded symbols received is close to Shannon's optimal, i.e. K encoded symbols, the average degree of each encoded symbols should be at least $\ln K$ for the sake of making the decoding possible. Consequently, it is necessary to ensure that $\sum_{i=1}^{K}[d\mu(d)] > \ln K$ when the parameters P_1, γ and A are selected for the RSF-LT codes.

3. Results and Discussions

In this section, the characteristics and performance of (i) LT codes based on robust soliton degree distribution [1]; (ii) SF-LT codes [8]; (iii) LT code with decreasing ripple size in [7] and (iv) proposed RSF-LT codes are compared. The particular LT codes to be studied are as follows.

- Robust LT code: LT codes based on robust soliton distribution with parameters $\beta = 0.1$ and $\delta = 1$ have been proven to provide the smallest average overhead factor [10]. Such robust LT codes are used here.
- SF-LT1: SF-LT code using $P_1 = 0.1$ and $\gamma = 2.0$
- SF-LT2: SF-LT code using $P_1 = 0.09$ and $\gamma = 2.1$
- RSF-LT1: RSF-LT code using $P_1 = 0.1$ and $\gamma = 1.9$
- RSF-LT2: RSF-LT code using $P_1 = 0.1$ and $\gamma = 2.1$
- RSF-LT3: RSF-LT code using $P_1 = 0.1$ and $\gamma = 2.0$
- RSF-LT4: RSF-LT code using P_1=0.09 and $\gamma = 2.1$
- LT code in [7]: LT code with decreasing ripple set proposed in [7] using $n = 1075, R = 21$ when $K = 1024$; and $n = 2108, R = 25$ when $K = 2048$.

3.1. Theoretical Evolution of the Ripple size

The theoretical evolution of the ripple size for the proposed RSF-LT code is evaluated by substituting Eq.(3) into Eq.(2). The results are plotted, together with those of robust LT code, SF-LT codes and LT code in [7], in Fig. 2 when the number of input symbols $K = 1024$. It can be observed that except for the LT code in [7] which has a decreasing ripple size, the ripple sizes of other LT codes decrease initially, and then increase before decrease again.

The results in Fig. 2 also indicate that RSF-LT3 code and RSF-LT4 code can recover the 1024 input symbols when $(1+0.05)\times 1024 \approx 1075$ encoded symbols (i.e., overhead $\alpha = 0.05$) have been received. Furthermore, RSF-LT2 code can recover all input symbols when the number of received encoded symbols approaches $N = (1+0.03)\times 1024 \approx 1055$. For the robust LT code, the ripple set will become smaller than 1 during the evolution if the overhead factor equals $\alpha = 0.06$, implying that the decoding process will fail.

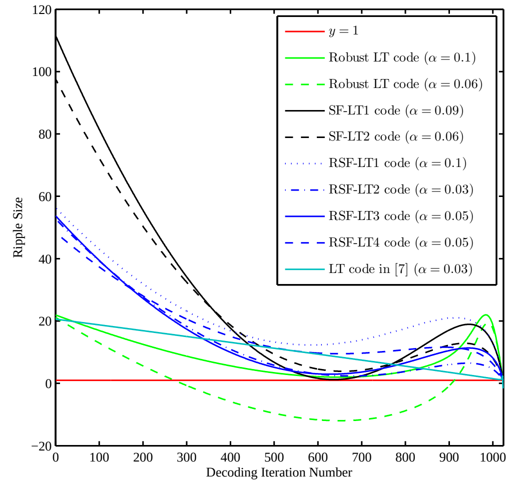

Figure 2: Theoretical ripple evolution of the Robust SF-LT codes, SF-LT codes and Robust LT codes when $K = 1024$.

With the same overhead $\alpha = 0.06$, SF-LT2 code can recover all encoded symbols. However, as shown in Fig. 1, SF-LT2 code cannot recover the encoded symbols if the overhead is reduced to $\alpha = 0.05$. For LT codes in [7], the ripple set will not become empty when overhead $\alpha \geq 0.03$. In summary, the results indicate that theoretically the proposed RSF-LT codes can achieve similar or even smaller overhead factor compared with other types of LT codes.

Note that in the iterative decoding process, not all the input symbols connected to a newly released degree-1 encoded symbol will useful. The reason is that they may already exist in the ripple set. Thus, the actual number of encoded symbol ($(1 + \alpha) \times K$) required to recover K input symbols may be larger than the results revealed in this section. Yet, the theoretical ripple evolution can provide an effective method for selecting parameter sets for RSF-LT codes.

3.2. Code Characteristics

The characteristics of the LT codes described in the previous section is further studied over a perfect channel. For each of the LT codes, $M = 2000$ different sets are constructed and evaluated when $K = 1024$ and $K = 2048$. The following symbols are defined.

- $\bar{d}$: average degree of the encoded symbols
- $\bar{x}$: average number of XOR operations for generating an encoded symbol
- $\bar{\varphi}$: average number of XOR operations for decoding an LT code over a perfect channel
- $\bar{\alpha}$: average overhead factor.

Table 1 lists the characteristics of the LT codes under study. It can be observed that when $K = 1024$ and 2048, all the proposed RSF-LT codes outperform the robust LT code in terms of average overhead factor $\bar{\alpha}$ and average number of encoding/decoding operations ($\bar{x}$ and $\bar{\varphi}$). The proposed RSF-LT codes also (i) outperform the SF-LT codes in terms of average overhead factor and achieve similar range of average number of encoding/decoding operations compared with the SF-LT codes; and (ii) outperform the LT code in [7] in terms of the average number of encoding/decoding operations and achieve similar range of average overhead factor compared with LT code in [7].

3.3. Decoding Performance over a BEC

The decoding performance of the LT codes over a BEC is simulated with an erasure probability of $P_{era} = 0.1$. Fig. 3 and Fig. 4 plot the probability of successful decoding of the LT codes when $K = 1024$ and $K = 2048$, respectively.

In Fig. 3 and Fig. 4, it can be observed that all the proposed RSF-LT codes and the SF-LT2 code outperform robust LT code when the number of encoded symbols received is relatively small (less than 1220 at $K = 1024$, 2300 at $K = 2048$). When the number of encoded symbols received become large, robust LT code begins to outperform other codes but is still outperformed by RSF-LT1 code and RSF-LT3 code. Compared with LT code in [7], RSF-LT3 code achieves a similar probability of successful decoding when the number of encoded symbols received is relatively small. As the number of encoded symbols received becomes larger, RSF-LT1 code and RSF-LT3 code outperform LT code in [7]. Moreover, RSF-LT2 code and RSF-LT4 code achieve a similar probability of successful decoding with LT code in [7] with the same number of encoded symbols received. Based on the results listed in Table I, it can be further concluded that both RSF-LT1 and RSF-LT3 codes can achieve the best performance in terms of encoding/decoding complexity ($\bar{x}$ and $\bar{\varphi}$) as well as probability of successful decoding when $K = 1024$ and $K = 2048$.

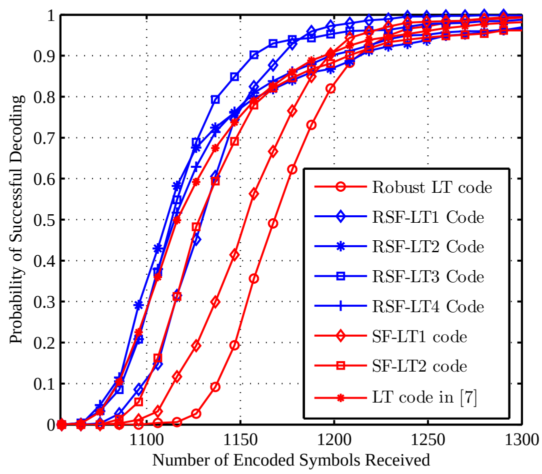

Figure 3: Probability of successful decoding versus the number of encoded symbols received. $K = 1024$ and $P_{era} = 0.1$.

Table 1: Code characteristics

K	Code	$\bar{d}$	$\bar{x}$	$\bar{\varphi}$	$\overline{\alpha}$	K	Code	$\bar{d}$	$\bar{x}$	$\bar{\varphi}$	$\overline{\alpha}$
1024	Robust LT code	9.94	8.94	10722	0.155	2048	Robust LT code	11.15	10.15	23729	0.129
	RSF-LT1 code	9.57	8.59	9935	0.115		RSF-LT1 code	10.77	9.74	22067	0.099
	RSF-LT2 code	7.49	6.49	7562	0.120		RSF-LT2 code	8.11	7.09	16339	0.110
	RSF-LT3 code	8.35	7.34	8241	0.107		RSF-LT3 code	9.18	8.12	18155	0.085
	RSF-LT4 code	7.53	6.50	7576	0.113		RSF-LT4 code	8.14	7.14	16266	0.099
	SF-LT1 code	9.20	8.19	9655	0.135		SF-LT1 code	10.16	9.14	21232	0.123
	SF-LT2 code	7.54	6.54	7671	0.125		SF-LT2 code	8.08	7.07	16374	0.117
	LT code in [7]	15.24	14.20	16346	0.118		LT code in [7]	17.51	16.61	37689	0.098

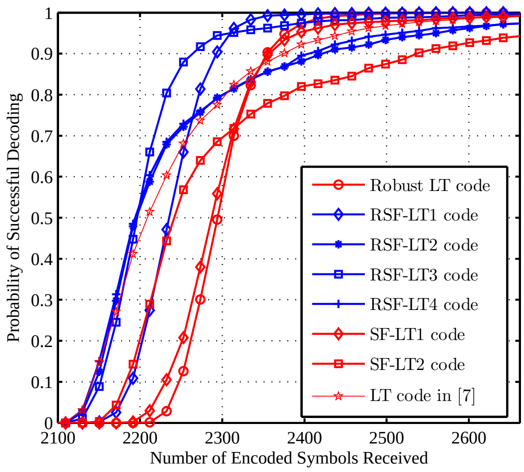

Figure 4: Probability of successful decoding versus the number of encoded symbols received. $K = 2048$ and $P_{era} = 0.1$.

4. Conclusion

In this paper, a new type of LT code called robust scale-free LT (RSF-LT) code has been proposed. It integrates the characteristics of (i) LT codes based on ideal soliton distribution and (ii) scale-free LT codes. Theoretical analyses on the ripple evolution process indicate that the proposed RSF-LT code outperforms robust LT code, SF-LT code and LT code in [7]. Simulations over a BEC further reveals that among all the LT codes under study, RSF-LT1 code and RSF-LT3 code achieve the best probability of successful decoding as well as the lowest encoding/decoding complexity.

Acknowledgment

This research was supported by the National Natural Science Foundation of China (Grant Nos. 61372095,61374178,61402092), The online education research fund of MOE research center for online education, China (Qtone education, Grant No.2016ZD306) and the Ph.D. Start-up Foundation of Liaoning Province, China (Grant No. 201501141).

References

[1] M. Luby, "LT codes", *43rd Annual IEEE Symposium on Foundations of Computer Science*, Vancouver, Canada, pp. 271–282, 2002.

[2] J. Yue, Z. H. Lin, B. Vucetic, G. Q. Mao, T. Aulin, "Performance Analysis of Distributed Raptor Codes in Wireless Sensor Networks", *IEEE Trans. on Communications*, vol. 61, no. 10, pp. 4357–4368, 2013.

[3] J. Abouei, J. D. Brown, K. N. Plataniotis, S. Pasupathy, "On the Energy Efficiency of LT Codes in Proactive Wireless Sensor Networks," *IEEE Trans. on Signal Processing*, vol. 5, no. 3, pp. 1116–1127, 2011.

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

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

[6] H. P. Zhu, G. X. Zhang, G. X. Li, "A novel degree distribution algorithm of LT codes," *Proceedings of IEEE ICCT*, Hangzhou, China, pp. 221–224, 2009.

[7] J. H. Sørensen, P. Popovski, J. Østergaard, "Design and Analysis of LT Codes with Decreasing Ripple Size," *Arxiv preprint PScache/1011.2078v1*, 2010.

[8] Y. L. Zhao, F. C. M. Lau, Z. L. Zhu and H. Yu, "Scale-free Luby Transform Codes," *International Journal of Bifurcation and Chaos*, vol. 22, no. 4, 1250094 (11 pages), 2012.

[9] J. H. Sørensen, P. Popovski, J. Østergaard, "On LT codes with decreasing ripple size," *Proceedings of ITA Workshop*, La Jolla, USA, 2011.

[10] U. Frank, X.X Hua, A.C Andrew, "Evaluation of a high performance erasure code implementation," *Technical Report*, University of California, San Diego, USA, 2004.