American Journal of Applied Sciences 11 (9): 1584-1593, 2014
ISSN: 1546-9239

doi:10.3844/ajassp.2014.1584.1593 Published Online 11 (9) 2014 (http://www.thescipub.com/ajas.toc)

THROUGHPUT AND BIT ERROR RATE ANALYSIS OF LUBY TRANSFORM CODES WITH LOW AND MEDIUM NODAL DEGREE DISTRIBUTIONS

[1]I. Joe Louis Paul, [2]S. Radha and [3]J. Raja

[1]Department of Information Technology,
[2]Department of Electronics and Communication Engineering,
SSN College of Engineering, Chennai, India
[3]Department of Electronics and Communication Engineering,
Sri Sai Ram Engineering College, Chennai, India

Received 2014-03-06; Revised 2014-03-06; Accepted 2014-08-07

ABSTRACT

This study presents two degree distributions namely low and medium nodal degree distributions aiming to build a low overhead Luby Transform (LT) codes. The motivation is to design a fast encoder/decoder especially for real-time multimedia streaming and multicasting applications using LT codes. The key idea of this study is to restrict the average degree of the transmitted encoded symbols as minimal. The impacts of low and medium degree encoded symbols on the performance of LT codes over an Additive White Gaussian Noise channel (AWGN) have been analyzed by the means of Bit Error Rate (BER), encoder/decoder delay, ripple size, throughput, overhead and bandwidth utilization as the performance metrics. Simulation results show that the proposed nodal degree distributions for LT codes achieve better throughput and BER performance at low overhead and delay with minimal decoding iterations by having a constantly decreasing ripple in comparison with conventional Robust Soliton Distribution (RSD) based LT codes.

Keywords: LT Codes, Low and Medium Nodal Degree Distributions, Additive White Gaussian Noise (AWGN) Channel, Bit Error Rate (BER)

1.INTRODUCTION

Luby Transform (LT) codes are the first realization of digital Fountain codes (also called as rateless codes), specifically designed for erasure channels to transmit the data reliably. LT codes are rateless in the sense that an infinite stream of encoded message symbols are being transmitted until the decoder reconstructs all the original message symbols K (Luby, 2002). Here, the message symbols are decoded from any subset ofencoded message symbols N, slightly larger than K. Recently, LT codes find its suitability in many applications due to its simple and efficient, exclusive OR (XOR) based implementations of both encoder and decoder (Byers *et al*., 1998; Cataldi *et al*., 2009).

For achieving reliable data delivery in cellular mobile and wireless broadcasting applications, the Automatic Repeat Request (ARQ) scheme may not be an appropriate one to use. Because, ARQ allows the retransmissions of data, introducing more delay, which is not acceptable in broadcasting applications (Eduardo *et al*., 2010). Low-Density Parity-Check (LDPC) codes are one of the Forward Error Correction (FEC) schemes achieve reliable communication with minimum retransmissions (Khedr and Sharkas, 2012). However, the assumption is that both transmitter and receiver should know the prior knowledge about channel conditions. This may not be feasible like in Internet, where the channel condition is always time-varying. Adaptive coding is one of the effective mechanisms to achieve the

Corresponding Author: I. Joe Louis Paul, Department of Information Technology, SSN College of Engineering, Chennai, India
Tell: 044-27469700; Fax: 044-27469772

maximum throughput in time varying channel conditions (Sekar *et al*., 2011). Hence, LT codes prove to be an ideal choice for these applications due to its adaptive nature to the varying channel conditions.

In LT codes, a data stream to be transmitted is divided into K blocks of bits known as source symbols or message symbols with fixed length. LT encoder takes K source symbols as an input and generates N encoded symbols where N is slightly larger than K based on the underlying degree distribution (Cataldi *et al*., 2006). On the basis of degree distribution Ω(d), LT encoder determines the degree d_i of each encoding symbol where i varies from 1 to N. Here, LT encoder uses simple XOR operations to construct each encoding symbol independent of other encoding symbols. The continuous stream of encoded symbols is transmitted over the communication channel. At the receiver, N consecutively received encoded symbols are collected by LT decoder for reconstructing K' source symbols (where $K' \leq K$) as illustrated in **Fig. 1**.

Although LT codes have the advantages of being simple and fast compared to other traditional coding schemes such as block codes and convolutional codes but may cause a bottleneck in terms of bandwidth utilization. The value of N, i.e., the number of encoded symbols plays an important role in the performance of LT codes. If N is large, then LT decoder achieves better throughput (i.e., successful recovery of source symbols) at the cost of encoding/decoding overhead involving more computations and consumes more bandwidth. For smaller value of N, LT decoder terminates prematurely with some source symbols yet to be recovered causing minimal throughput (i.e., poor success rate) but uses minimal bandwidth and low encoding/decoding overhead.

Hence, determining the overhead of LT codes is the key design criteria to achieve an optimal balance between throughput and bandwidth. This issue can be addressed in two ways: (i) Determining number of encoded symbols required for recovering all source symbols and (ii) determining the number of source symbols that can be recovered for the specific number of encoded symbols. The second approach finds the suitability of LT codes for the limited channel conditions such as wireless.

The conventional LT codes achieve the maximum throughput only by increasing the required transmission bandwidth which may not be applicable if the channel is band-limited. The aim of this study is to reduce the bandwidth requirement of LT codes at the same time achieving the same throughput at low bit error rate. This becomes feasible only if the degree distribution function used at the LT encoder is optimal.

While, traditional degree distribution functions originally designed by Luby for LT codes performs better, but still the researchers are finding their own new ways to optimize the degree distribution and its impact on the performance of LT codes over various channel conditions.

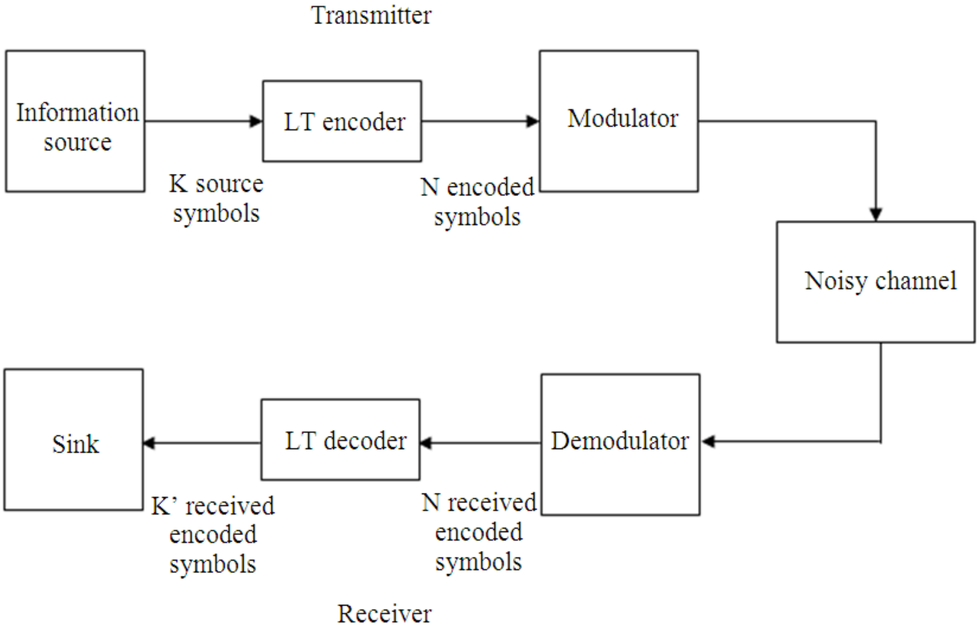

Fig. 1. LT encoding and decoding

Hence, the main focus of the various degree distribution functions discussed in literature for LT codes is to design a more efficient decoder (Jenkac *et al*., 2005). But, the decoding efficiency of LT code directly depends on the overhead involved in the underlying encoding process at the transmitter. Because, the degree distribution ensures that LT decoder recovers K source symbols from N received encoded symbols with high probability (where N is slightly larger than K).

At the same time, when the encoded symbols are transmitted over the channel, there is a probability of errors being introduced into the system and these errors might affect the integrity of the system. Hence, it becomes essential to assess the overall performance of LT codes with Bit Error Rate (BER) as a key parameter. One approach that can be used to reduce the BER is to reduce the bandwidth. But, this results in reduction in throughput of the system. Therefore, the motivation for this study is to adopt low and medium nodal degree distribution functions for LT codes and to determine the optimal performance with restricted maximum nodal degree that aims to achieve the maximum throughput with smaller N.

The remainder of the paper is organized as follows. In section 2, we briefly introduce the various degree distribution functions already proposed for LT codes and the need for its optimization. Section 3 describes the proposed degree distributions for LT codes. Section 4 deals with simulation results. The summary of our findings are discussed in section 5. Finally, we give our conclusions in section 6.

2. RELATED WORK

The performance of LT codes depend on a given degree distribution. So, this section discusses the various degree distribution functions that were earlier proposed in literature for LT codes.

The initial work by Luby on Ideal Soliton Distribution (ISD) promised to achieve the lower bound on encoding/decoding overhead by maintaining a constant ripple of size one during each decoder iteration, may be having some redundant degree 1 encoded symbols.

But practically, the poor design of random degree generator, as a part of LT encoder makes the ISD based conventional LT code to suffer by premature termination of decoding. This is due to the absence/less number of degree 1 encoded symbols and/or non-selection of some of the source symbols as the neighbors in any of the encoded symbols generated (MacKay, 2005).

Hence, the Robust Soliton Distribution (RSD), the variant of ISD was also proposed by Luby, promising that there will be always more than one degree 1 encoded symbols in the ripple during decoding iteration. So that, the probability of successful recovery of source symbols by the decoder can be increased. But, the decoder overhead increases exponentially as the number of source symbols K increases.

In continuation to that, Raptor codes are also fountain codes built upon LT codes, invented by (Shokrollahi, 2006) mainly to address the non-linear decoding property of LT codes. To ease the recovery process and fast encoding/decoding, Raptor codes employ the pre-coding as the outer code and concatenating with LT code to achieve linear time encoding and decoding by having minimal average degree d_{min_avg} of the encoded symbol compared to LT codes.

Hyytiä *et al*. (2006) emphasized the need for designing the proper degree distribution for LT codes in their work for optimizing the number of encoded symbols required for achieving the maximum decoding probability (Hyytiä *et al*., 2007). Sanghavi (2007) also investigated the intermediate performance of LT codes for the limited number of received encoded symbols at the decoder, especially for real-time scenarios where users do not receive sufficient number of output symbols. Bodine and Cheng (2008) discussed the importance of having smaller number of encoded symbols by optimizing various parameters of the Robust Soliton distribution to reduce the encoder/decoder delay and to maximize the throughput. The Suboptimal Degree Distribution (SODD) for LT codes for improving efficiencies of data distribution applications was presented (Zhu *et al*., 2008; 2009).

In addition to that (Chen *et al*., 2010) also proposed the evolutionary computation techniques for optimizing the degree distribution used in LT codes. Zang and Feng (2011) also analyzed the two commonly used distributions Ideal Soliton and Robust Soliton degree distributions and found that the number of degree 1 encoded symbols play a vital role not only in the successful decoding of source symbols and also in deciding the overhead of the encoder/decoder. Sorensen *et al*. (2012) also emphasized the need for decreasing ripple size during decoding which reduces the decoding overhead.

Zhiliang *et al*. (2012) introduced different metrics like average degree, release probability and overhead to analyze the performance of LT codes and proposed a well defined degree distribution for LT codes. Hence, the motivation behind this study is to achieve the optimal performance of LT codes by successfully recovering all source symbols at low bit error rate with minimal delay and overhead by proposing low and medium nodal degree distributions.

3. PROPOSED WORK

This section illustrates the LT encoding process as a bipartite graph, the need for modifying the degree distribution and followed by the proposed degree distributions for LT codes.

3.1. Bipartite Graph Representation of LTEncoding

A message is a stream of data that consists of bits. This stream of data is partitioned into K source symbols represented as S = {s_1, s_2, s_3, ..., s_K}, where the symbol length is same for all K source symbols. The LT encoder accepts these K source symbols as the input and produces an infinite stream of encoded symbols or codewords by the use of an encoding algorithm.

This algorithm generates an encoded symbol e_i by performing XOR operations on randomly and uniformly selected d_i source symbols, where d_i is the randomly chosen degree based on the degree distribution Ω(d) for the encoded symbol e_i from the degree sequence D = {d_1, d_2, d_3, ….. d_K}.

The degree d_i decides the number of unique source symbols that can be chosen as the neighbors to construct an encoded symbol e_i. The connection between the source symbols and the encoded symbols can be modeled as the bipartite graph G as described in **Fig. 2**.

Figure 2 illustrates the LT encoding process as the bipartite graph where the number of vertices in V_1 and V_2 are K and N, where K and N are the number of source symbols and the number of transmitted encoded symbols respectively.

3.2. Analyzing the Role of Modifying Degree Distribution

The connections between the encoded and source symbols of the bipartite graph G shown in **Fig. 2** can reveal the complex patterns for the increase in number of source symbols that may lead into a complex structure of the bipartite graph. Therefore, analyzing the complex bipartite graph is a quite challenging one for understanding the LT encoder process. Hence, the design of the LT encoder can be viewed as a simplified process by modifying the degree distribution.

At the same time, the careful design of the degree distribution function Ω(d) decides the complexity of both LT encoding and decoding processes. Because, the degree of each encoded symbol generated by LT encoder varies from 1 to K. The maximum degree d_{max}, an encoded symbol can hold is K called higher degree symbol.

In LT codes, there should be enough number of higher degree encoded symbols to ensure that all source symbols participate in the encoding process. This helps to recover as many number of source symbols as possible. At the same time, the number of higher degree symbols must be controlled because they increase the computational complexity of both the encoder and decoder.

Hence, there is a need for as many number of lower degree symbols (where d is 1 or 2) to make the decoder to run continuously. This maintains a constant ripple for the decoder to continue its further recovery of source symbols.

Therefore, a good degree distribution should ensure that there is always a balanced number of lower and higher degree encoded symbols generated by the encoder. Therefore, the average degree d_{avg} of an encoded symbol is bounded as log K. Now, the lower bound on the number of encoded symbols N_l can be determined as K multiplied by d_{avg}.

Hence, the objective of this proposed work is to present the two simplified degree distribution schemes for LT encoder and to determine the suitability of both schemes for the transmission of encoded symbols over Additive White Gaussian Noise (AWGN) channel to achieve the better performance of LT codes in terms of performance metrics such as BER, delay, constantly decreasing ripple, overhead, throughput and bandwidth utilization.

3.3. Low Degree Distribution (LDD)

The significance of the degree distribution function used at the LT encoder side for the successful decoding of all the source symbols is understood by experimenting the LT code with the different degree combinations of encoded symbols like {degree 1, degree 2}, {degree 1, degree 3}, {degree 1, degree 4} and etc. It was found that, the combination of degree 1 and degree 2 encoded symbols achieves the better performance in terms of bandwidth utilization, overhead and delay in comparison with other combinations.

In this proposed scheme, all the encoded symbols have only *lower degrees* with the degree as either 1 or 2 as similar in real-time networks. The probability of choosing encoded symbols having degree 1 is same as that of the probability of encoded symbols having degree 2. So, the maximum degree of this distribution d_{max} is only two. That is, the degree distribution has been restricted in such a way that, there is an optimal balance between the number of degree 1and degree 2 encoded symbols.

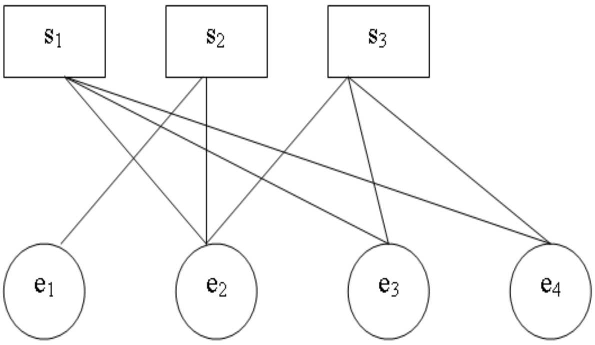

Fig. 2. Bipartite graph representation ofLT encoding

Depending on the number of encoded symbols to be transmitted, the fraction of degree 1 and degree 2 encoded symbols vary. The purpose of this distribution scheme is to reduce the complexity of the encoder/decoder operations, delay, ripple size and overhead. The performance of the same has been explained in the simulation results.

3.4. Medium Degree Distribution (MDD)

In traditional random networks, most nodes have a medium node degree. The degrees of all nodes are distributed around the average. Hence, in this proposed scheme, the degree of the encoded symbols has been considered only in the combinations of {degree1, degree 2, degree 3, degree 4}. Here, the degree distribution is restricted in such a way that, there is an optimal balance between the distributions of degree 1 degree 2, degree 3 and degree 4 encoded symbols.

This distribution is actually a mixture of lowerand medium degree encoded symbols in contrast to LDD. Here, the maximum degree d_{max} is restricted to 4 in order to reduce the encoding/decoding overhead. The performance of the proposed MDD based LT codes is investigated for the given input source symbols by sending the slightly larger number of encoded symbols for recovery. That is, the number of encoded symbols that are needed for recovery is well-defined. In this model, only N_l.

4. SIMULATION RESULTS

The above described methods are simulated with the following specifications. Sample source data taken for analysis is 10^6 bits. The number of source symbols is 100 with the symbol length of 10,000 bits. The performance of LT codes using LDD, MDD and RSD are studied over AWGN channel with Binary Phase Shift Keying (BPSK) as the modulation scheme by varying the number of Encoded Symbols (ES) to be transmitted from 100 to 300 with a step size of 50.

In addition, RSD has been implemented with the number of input source symbols K = 100 and tested for varying the failure probability of the LT process δ and a positive constant c that affects the probability of generating degree 1 encoded symbols. In the simulation analysis for RSD, $\delta = 0.05$ and $c = 0.2$ have been used.

4.1. BER Vs SNR

The BER performance of LT codes based on LDD, MDD and RSD over the AWGN channel is presented in **Fig. 3**.

Figure 3 shows BER Vs Signal-to-Noise Ratio (SNR) performance of LT codes for the number of encoded symbols ES = 200 with the respective Throughput (T) achieved, that is the number of source symbols that are successfully recovered at the decoder.

4.2. Average Encoder/Decoder Delay

The average encoder and decoder delays are considered as the two performance metrics. Here, the average encoder delay is the time taken by the LT encoder to generate the required number of encoded symbols, where the average decoder delay is the time taken for the decoder to recover the source symbols from the received encoded symbols. The delay performance is plotted for the number of encoded symbols varied from 100 with a step size of 50 with respect to SNR.

Figure 4 and **5** show the average encoder/decoder delay (in msec) experienced by LDD, MDD and RSD based LT codes as a function of the number of encoded symbols. As the number of encoded symbols increases, the average encoder/decoder delay also linearly increases in all the three distributions.

4.3. Constantly Decreasing Ripple Size

The successful recovery of original source symbols K by LT decoding process is truly depending upon the key parameter called ripple size. The ripple is a buffer used in decoding to store the count of degree 1 encoded symbols by including these symbols itself. The modified degree distribution ensures that a constantly decreasing ripple size is maintained throughout the decoding process. Based on the number of encoded symbols generated, a desired ripple size could be maintained at the decoder. And also, the overhead of the decoder can be determined for its successful termination. **Figure 6** illustrates the decreasing ripple size during decoding process for the varying number of Encoded Symbols size (ES), using LDD, MDD and RSD respectively.

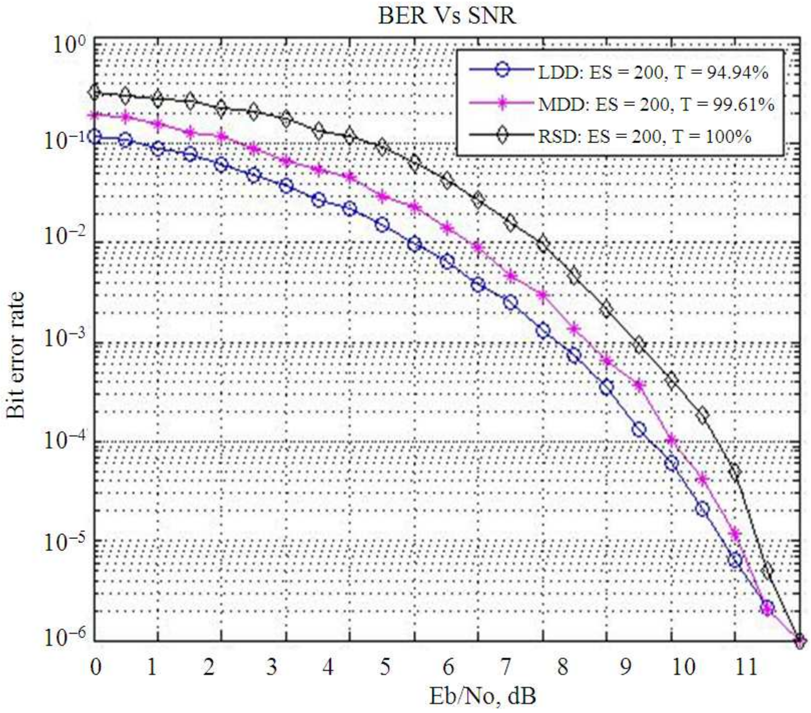

Fig. 3. BER Vs SNR performance of LDD (ES = 200), MDD (ES = 200) and RSD (ES = 200) based LT codes

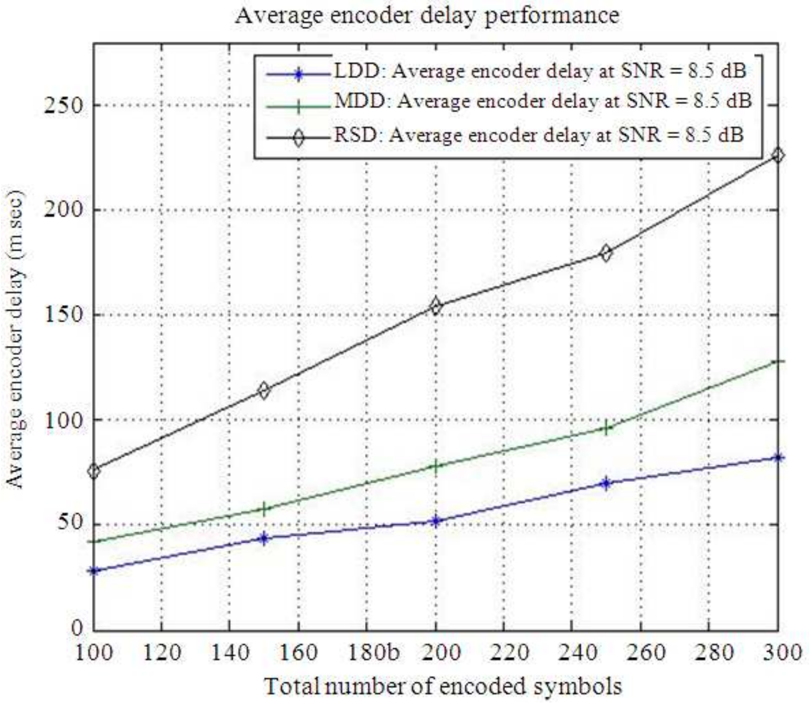

Fig. 4. Average encoder delay performance of LDD, MDD and RSD based LT codes

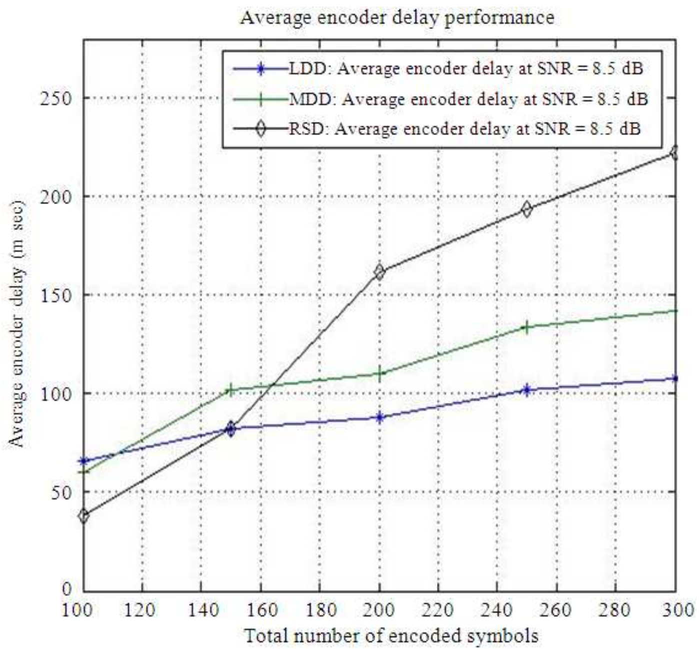

Fig. 5. Average decoder delay performance of LDD, MDD and RSD based LT codes

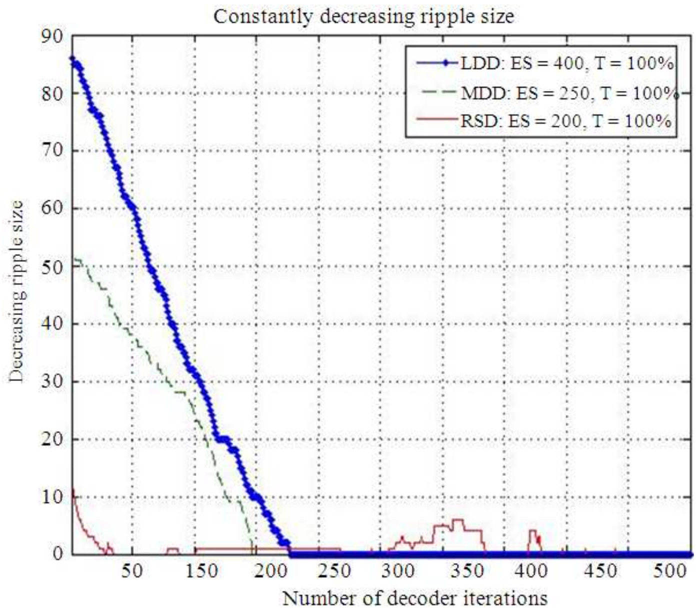

Fig. 6. Constantly decreasing ripple size for LDD, MDD and RSD

4.4. Throughput

The total number of encoded symbols transmitted by the sender decides the successful termination of the decoder. Therefore, the LT decoder successfully terminates only if all the source symbols are recovered. Hence, the throughput of the LT codes T is measured as the ratio between the number of source symbols successfully recovered by the decoder $K^{'}$ with respect to total number of source symbols K. **Figure 7** shows the comparative throughput performances of LDD, MDD and RSD based LT codes for the varying number of encoded symbols.

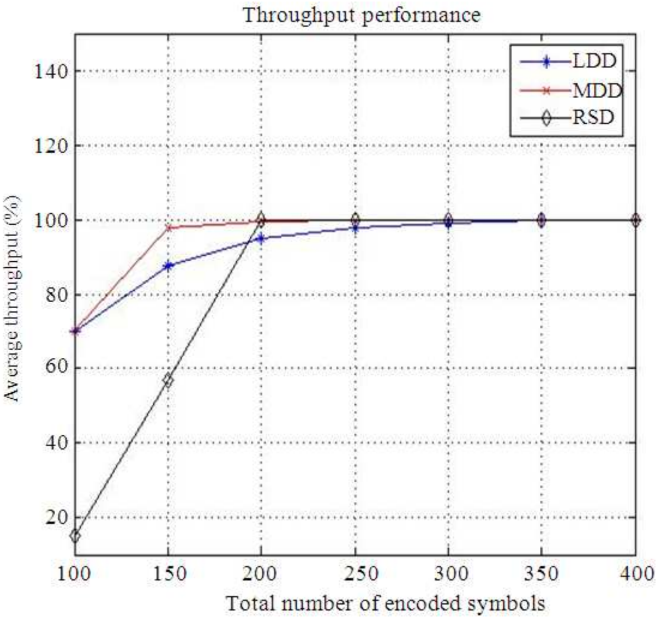

Fig. 7. Throughput performance of LDD, MDD and RSD based LT codes

5. DISCUSSION

The BER Vs SNR plot as shown in **Fig. 3** illustratesthat LDD based LT code gives approximately 0.5 dB improvements over MDD and RSD based LT codes at the cost of 6% unrecovered source symbols. The BER achieved by using LDD and MDD is 10^{-4} at SNR=8.5 dB whereas by using RSD, the BER is 10^{-3} with 100% recovered source symbols.

Figure 3 clearly shows that by using the proposed schemes in LT codes, approximately $\geq 10^2$ bits in error, for the given message of 10^6 bits. This motivates us to find out the required SNR for achieving the error-free transmission using LT codes over AWGN channel using the proposed schemes in comparison with RSD. It is clearly found that improved BER performances of 10^{-4} to 10^{-6} is obtained by increasing the SNR from 8.5 dB by 2 dB in LDD whereas for both MDD and RSD, it requires an addition of 0.5 dB more SNR than LDD to achieve the same error-free performance. This improvement in BER performances has been achieved by an additional signal power of 1.58 mw by using LDD and 1.77 mw for MDD and RSD approximately.

The encoder delays of LT codes based on all the three distribution methods increases by an average of approximately 32% as shown in **Fig. 4**. Hence, the encoder delay is linear with respect to the number of encoded symbols. And also, it is found that LDD gives minimum encoder delay compared to other two at the encoded symbols size ES = 300. **Figure 4** also illustrates that the proposed MDD and LDD schemes decreases the average encoder delay by about 43% to 64% respectively compared with RSD for the number of encoded symbols ES = 300.

Figure 5 clearly proves that the average decoder delay is also linear with respect to number of encoded symbols for all the three distributions. However, the linearity constant is different for each distribution. It is found that, the decoder delay increases by an average of approximately 13, 26 and 62% respectively for every addition of 50 more encoded symbols using LDD, MDD and RSD.

Unlike encoder delay, the rate of change in decoder delay varies by varying the number of encoded symbols in LDD, MDD and RSD. And these variations are small in LDD and MDD compared to RSD. **Figure 5** also explains that the proposed MDD and LDD schemes decreases the average decoder delay by about 36 to 51% respectively compared with RSD for the number of encoded symbols ES = 300.

Figure 6 illustrates the decreasing ripple size during decoding process for LDD, MDD and RSD respectively. Smaller the ripple size implies only the fraction of the K source symbols can be successfully recovered. That is, throughput of the system greatly influenced by the ripple size. Hence, designing an optimal ripple size is a major concern for achieving higher throughput and also to keep the decoding overhead in under controlled. Hence, rather than maintaining a constant ripple size, the proposed degree distributions sustain a constantly decreasing ripple size for the successful termination of the decoder. The constantly decreasing ripple size for LDD, MDD and RSD for maximum throughput is described in **Fig. 6**.

The decoder successfully terminates only if the ripple size is zero with the number of recovered source symbols K'= K. On observing the results from **Fig. 6**, the convergence point for ripple size reaching zero with maximum successful decoding is determined for LDD, MDD and RSD. **Figure 6** clearly reveals that, the MDD reaches the convergence of ripple size for successful decoding very quickly than LDD and RSD.

The number of decoder iterations required for MDD to achieve the convergence of ripple size to zero is 17 to 64% lower compared with LDD and RSD respectively. Then, the optimal size for the initial ripple using MDD is to have nearly 20% of the encoded symbols as the degree 1 encoded symbols for achieving the optimal performance of LT codes.

It is inferred that by varying the number of encoded symbols, the throughput T can be increased as shown in **Fig. 7**. In LDD, since the selection of source symbols to be encoded allows redundancy, the maximum throughput performance can be achieved by consuming 4 times of the bandwidth. In MDD, the maximum throughput is achieved by consuming 2.5 times of the bandwidth itself. Whereas using RSD, it requires only twice as much bandwidth. It is found that, in terms of bandwidth utilization, RSD performs better by reducing the bandwidth requirement by 20 to 50% compared with MDD and LDD respectively as shown in **Fig. 7**. These results clearly show that, the proposed degree distribution schemes for LT codes outperforms RSD based LT codes in terms of BER, delay, constantly decreasing ripple and overhead for successful decoding (Zhiliang *et al*., 2012).

6. CONCLUSION

In this study, two nodal degree distribution schemes such as LDD and MDD have been proposed to overcome the encoder/decoder overhead and delay by restricting the average degree of the encoded symbols as minimal. Simulation results show that considering BER as the metric, LDD gives the better error performance for the average transmission of 200 encoded symbols against MDD and RSD. In addition, LDD also minimizes the average encoder/decoder delay compared to other two schemes. Though, LDD seems to be better in terms of delay constraints, but it tries to extend the available bandwidth by 4 times. But, in terms of ripple, MDD reaches the convergence quickly than LDD and RSD during decoding. However, RSD outperforms both LDD and MDD in terms of throughput by consuming less bandwidth but at the cost of large encoder/decoder overhead and delay. In real-time multimedia streaming and multicasting applications, bandwidth and delay are the two primary concerns that need to be addressed. Hence, MDD based LT codes seems to be an ideal choice for reliable data transmission over noisy channel with tolerable encoder/decoder overhead, delay, BER, memory requirement in maintaining ripple and bandwidth conservation. Both LDD and MDD schemes can be further extended by analyzing the influence of varying the number of degree 1 encoded symbols on the performance of LT codes.

7. REFERENCES

Bodine, E.A. and M.K. Cheng, 2008. Characterizationof Luby transform codes with small message size for low-latency decoding.Proceedings of IEEE International Conference on Communications,May 19-23, IEEE Xplore Press, Beijing, China, pp:1195-1199. DOI: 10.1109/ICC.2008.233

Byers, J.W., M. Luby, M. Mitzenmacher and A. Rege,1998. A digital fountain approach to reliable distribution of bulk data. ACM SIGCOMM Comput. Commun. Rev., 28: 56-67. DOI: 10.1145/285237.285258

Cataldi, P., M. Gerla and F. Zampognaro, 2009. Ratelesscodes for file transfer over DVB-S. Proceedingsof 1st International Conference on Advances in Satellite and Space Communications, Jul. 20-25, IEEE Computer Society, Los Alamitos, California, pp: 7-12. DOI: 10.1109/SPACOMM.2009.20

Cataldi, P., M. P. Shatarski, M. Grangetto and E. Magli,2006. Implementation and performance evaluation of LT and raptor codes for multimedia applications. Proceedings of 2nd International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Dec. 18-20, IEEE Computer Society, Pasadena, California, USA, pp: 263-266.DOI: 10.1109/iih-msp.2006.264994

Chen, C., Y. Chen, T. Shen and J.K. Zao, 2010. On theoptimization of degree distributions in LT code with covariance matrix adaptation evolution strategy. Proceedings of IEEE Congress on Evolutionary Computation, Jul. 18-23, IEEE Xplore Press, Barcelona, Spain, pp: 1-8.DOI: 10.1109/CEC.2010.5586202

Eduardo, L.D., S.E.P Hernandez, G.R. Gomez, M.A. Medina and J.A.M. Gomez, 2010. An efficient causal protocol with forward error correction for mobile distributed systems. J. Comput. Sci., 6: 756-768.DOI: 10.3844/jcssp.2010.756.768

Hyytiä, E., T. Tirronen and J. Virtamo, 2006. Optimizing the degree distribution of LT codes with an importance sampling approach. Proceedings of RESIM, Oct. 9-10, Bamberg, Germany, pp: 64-73.

Hyytiä, E., T. Tirronen and J. Virtamo, 2007. Optimal degree distribution for LT codes with small message length. Proceedings of 26th IEEE International Conference on Computer Communications, May 6-12, IEEE Xplore Press, Anchorage, Alaska, pp: 2576-2580. DOI: 10.1109/INFCOM.2007.324

Jenkac, H., T. Mayer, T. Stockhammer and W. Xu, 2005. Soft decoding of LT-codes for wireless broadcast. Proceedings of IST Mobile Summit, Jun. 19-23, Dresden, Germany.

Khedr, M. and M. Sharkas, 2012. Wireless HD video transmission over multicarrier error-correction channels. J. Comput. Sci., 8: 1897-1913. DOI: 10.3844/jcssp.2012.1897.1913

Luby, M., 2002. LT codes. Proceedings of the 43rd Symposium on Foundations of Computer Science, Nov. 16-19, IEEE Computer Society, Washington, DC, USA, pp: 271-280.DOI:10.1109/SFCS.2002.1181950

MacKay, D.J.C., 2005. Fountain codes. Proc. IEEE Commun., 152: 1062-1068.DOI: 10.1049/ip-com:20050237

Sanghavi, S., 2007. Intermediate performance of rateless codes. Proceedings of Information Theory Workshop, Sep. 2-6, IEEE Xplore Press, California, USA, pp: 478-482.DOI: 10.1109/ITW.2007.4313121

Sekar, V., V. Palanisamy and K. Baskaran, 2011. Performance analysis of IEEE 802.16d using forward error correction. J. Comput. Sci., 7: 431-433.DOI:10.3844/jcssp.2011.431.433

Shokrollahi, A., 2006. Raptor codes. IEEE Trans. Inform. Theory, 52: 2551-2567. DOI: 10.1109/TIT.2006.874390

Sorensen, J.H., P. Popovski and J. Ostergaard. 2012.Design and analysis of LT codes with decreasing ripple size. IEEE Trans. Commun., 60: 3191-3197. DOI: 10.1109/TCOMM.2012.091112.110864

Zang, Q. and G. Feng, 2011. Degree distribution analysisof LT codes. J. China Univ. Posts Telecommun., 18: 28-33. DOI: 10.1016/S1005-8885(10)60214-2

Zhiliang, Z., L. Sha, Z. Jiawei, Z. Yuli and Y. Hai, 2012. Performance analysis of LT codes with different degree distribution. Proceedings of 5h International Workshop on Chaos-Fractals Theories and Applications, Oct. 18-21, IEEE Xplore Press, Liaoning, China, pp: 142 -146. DOI: 10.1109/IWCFTA.2012.39

Zhu, H., G. Zhang and G. Li, 2008. A novel degree distribution algorithm of LT codes. Proceedings of 11th IEEE International Conference on Communication Technology, Nov. 10-12, IEEE Xplore Press, Hangzhou, China, pp: 221-224. DOI: 10.1109/ICCT.2008.4716207

Zhu, H.P., G.X. Zhang and Z.D. Xie, 2009. Suboptimaldegree distribution algorithm of LT codes of digital fountain. J. Applied Sci. Electron. Inform. Eng., 27: 6-11.