Journal of Computer Science

Original Research Paper

Performance Analysis of Joint Degree Distribution (JDD) in Luby Transform Codes

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

[1]Department of Information Technology, SSN College of Engineering, Chennai, India
[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: 02-03-2014
Revised: 06-03-2014
Accepted: 07-08-2014

Corresponding Author:
Joe Louis Paul, I.
Department of Information Technology, SSN College of Engineering, Chennai, India
Email: joelouisi@ssn.edu.in

Abstract: Luby Transform (LT) codes, the first realization of rateless codes are widely used in wireless communication mainly for its adaptability to varying channel conditions and its capacity approaching performance. In spite of the above advantages and its simplicity in implementation, LT codes suffer from a bottle neck of premature termination due to the poor design of degree distribution. In this study, a novel degree distribution called Joint Degree Distribution (JDD) is proposed for the successful completion of LT encoder/decoder processes. The efficient utilization of the bandwidth is tried to be achieved by using only 'k' encoded symbols for 'k' source symbols, unlike in traditional systems. JDD is carefully designed to ensure that the encoding/decoding delay does not exceed that which is existent in the traditional systems. The performance of JDD for throughput and bit error rate, experimented over Additive White Gaussian Noise (AWGN) channel compared to conventional degree distribution was observed to be much better.

Keywords: LT Codes, Rateless Codes, Joint Degree Distribution, AWGN Channel

Introduction

One of the fundamental issues in any communication system is achieving the reliable delivery of data between a sender and a receiver over an unreliable channel. Automatic Repeat Request (ARQ) is an error control technique that allows the receiver to detect the errors in the received message and request the sender for the retransmission of the erroneous message. But, these message retransmissions causes more delay and may not be found suitable for cellular and real-time multimedia broadcasting applications (Eduardo *et al*., 2010).

As an alternate to ARQ, Forward Error Correction (FEC) schemes such as Low Density Parity Check (LDPC) codes introduce redundant data for achieving minimum retransmissions (Khedr and Sharkas, 2012). Here, the basic assumption is that the channel state information is known prior to both the transmitter and receiver. But, if channel conditions are time-varying especially like in Internet, then there will be more number of retransmissions of the message which may not be feasible for time-sensitive applications. Hence, it becomes necessary to achieve the maximum throughput over the time-varying channels also.

In literature, it is found that, adaptive coding techniques could exploit these time-varying channel conditions in order to optimize the performance of the communication system (Sekar *et al*., 2011). Therefore, Fountain codes have been an active research area for more than a decade in the communications field, primarily for its focus on achieving capacity approaching performance. Fountain codes also known as rateless erasure codes or simply rateless codes, are designed especially for Binary Erasure Channels (BEC). An infinite stream of encoded symbols is transmitted by the encoder to the decoder for recovery of the original source symbols, with the excess

encoded symbols exploiting the transmission bandwidth (Luby, 2002).

Though, rateless codes are initially designed for erasure channels for the efficient transmission of data, but these codes can also provide better performance over the time varying channels with increase in latency and complexity of the decoder. Unlike fixed length coding, there is no need for the transmitter to know the prior channel conditions before it starts transmitting the encoded symbols through the channel, if rateless codes are employed. This makes rateless codes essentially suitable for wireless channels also (Salkuyeh *et al*., 2013).

LT and Raptor codes are the two major classes of rateless codes. The capacity approaching performance and adaptability features of rateless codes make LT and Raptor codes as near optimal codes for erasure channels, Binary Symmetric Channels (BSC), Additive White Gaussian Noise channel (AWGN) and fading channels like Rayliegh and Rician (Palanki and Yedidia, 2004; Shokrollahi, 2006). But the above claim of achieving optimal state is feasible, only by applying an appropriate degree distribution function in LT encoder.

The basic principle of LT codes is that the receiver continuously receives an infinite stream of encoded symbols until the decoder recovers all the k source symbols from a set of n encoded symbols received, where n is slightly larger than k.

Here, every symbol is encoded "on the fly" based on the degree distribution d, randomly chosen for that particular symbol. The encoded symbol is obtained by performing simple bitwise exclusive-or (XOR) operations on randomly chosen and uniformly distributed d source symbols. The following Fig. 1 shows the system model used in Conventional Degree Distribution (CDD) based LT codes.

The source file with data to be transmitted is fragmented into 'k' source symbols of uniform length 'l' called as blocks or input symbols, where l >= 1. For a given set of k source symbols $\{s_1, s_2, \ldots s_k\}$ of the source file, the encoder (using LT codes) generates an infinite stream of encoded symbols $\{c_1, c_2, \ldots\ldots\}$.

The encoded symbols are actually obtained by XORing d randomly and independently selected source symbols based on the given degree distribution as described in Fig. 2. The receiver reconstructs the k source symbols from any subset of n encoded symbols where n is equal to or slightly larger than the number of source symbols k. In LT codes, the general assumptions made are (i) the length of symbols (both source and encoded) are of same length (ii) encoding process of the individual symbols are independent.

In addition to that, the LT decoder is designed to recover k original source symbols from k $(1+\varepsilon)$ encoded symbols, where $\varepsilon \geq 0$, is the encoding overhead. When the number of source symbols $k \to \infty$ and $\varepsilon \to 0$, the decoder behaves asymptotically optimal (Zao *et al*., 2012). When the encoded symbols overhead ε reaches its minimum, the number of encoded symbols n almost becomes equal to *k*. Hence, there is a marginal level of increase in the complexity of the decoder process which forces the premature termination of LT decoder.

Lu *et al*. (2009) addressed this issue and proposed a full rank decoding algorithm that extends the LT decoding process to avoid the premature termination, when there is no degree 1 symbol in the ripple to enable the decoder to continue processing. Sorensen *et al*. (2012) also have emphasized the need for decreasing ripple size during decoding which reduces the decoding overhead.

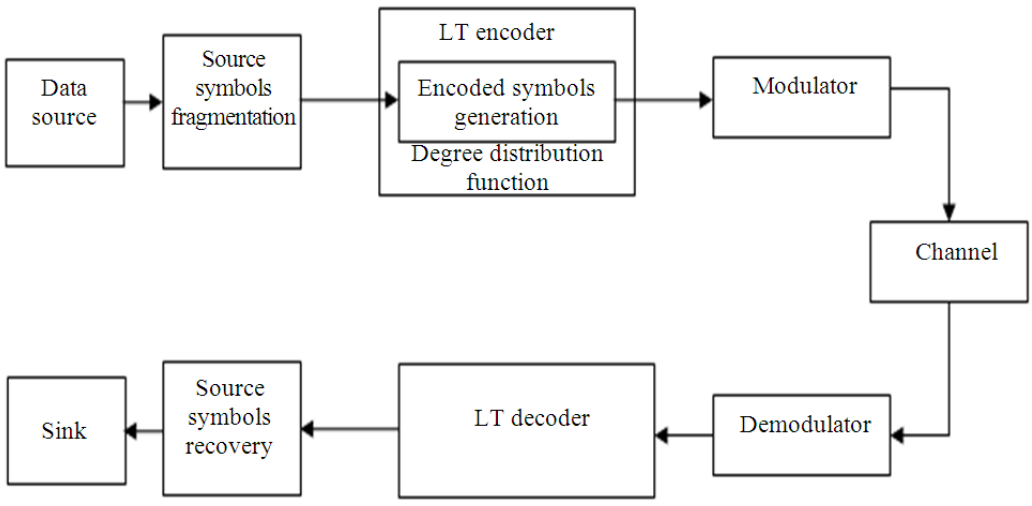

Fig. 1. Conventional degree distribution based LT codes system model

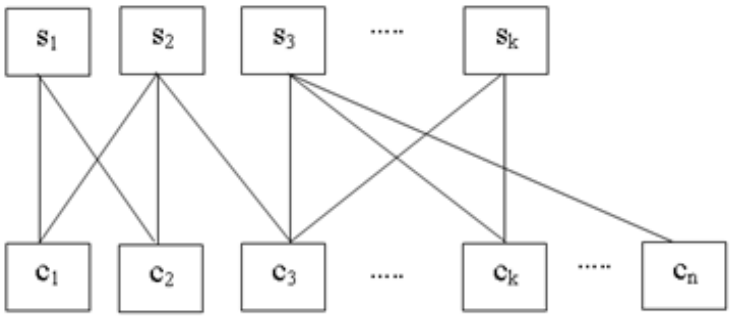

Fig. 2. LT encoding

Analogous to these experiments, our investigations are focused on analyzing the performance of LT codes for noisy channels. In this research work, a novel degree distribution function called Joint Degree Distribution (JDD) function is designed for the LT encoder. The focus of the proposed distribution strategy is to reduce the overhead of decoder and to achieve higher throughput with low encoding/decoding delays. It is also attempted to minimize the bit error rate which tends to increase with decrease in encoding overhead ε.

The rest 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 JDD based LT codes system model. Section 4 details about the simulation results. The summary of our findings are discussed in section 5. Finally, we give our conclusions in section 6.

Related Work

Motivation

The degree distribution is the key design part in LT codes. Luby (2002) has shown that, the optimal performance of LT codes could be achieved in his initial work with use of an Ideal Soliton Distribution (ISD) by maintaining the ripple size as constantly one during the entire LT decoding process. After the introduction of Robust Soliton Distribution (RSD), the decoding performance was slightly better as the constant ripple size was more than one.

Shokrollahi (2006) proposed a modified version of LT encoding scheme called Raptor codes having fixed rate precode, concatenated with LT code, where the input blocks are encoded using a fixed rate code before LT encoding. Raptor codes were designed especially to solve the transmission problems over the varying channel conditions.

Though, RSD proves to be performing well for larger number of encoded symbols, but it gives the motivation for the researchers to design an optimal degree distribution functions to reduce the number of encoded symbols being transmitted.

Zhu *et al*. (2009) proposed a Sub Optimal Degree Distribution (SODD) algorithm for LT codes in improving the efficiencies of data distribution applications and analyzed its drawbacks on realization with respect to the average decoding efficiency, the best decoding efficiency and the variance of decoding efficiencies were presented.

Chen *et al*. (2010) also introduced evolutionary algorithms to optimize the degree distribution in LT codes, so that degree distribution can be customized for different purposes. In continuation to that, Zang and Feng (2011) also analyzed Luby's ISD and RSD and proposed an improved adaptive encoding method to ensure the proper distribution of degree one encoded symbols for the successful decoding process. Zhiliang *et al*. (2012) proposed an approach to design a well defined degree distribution for LT codes and analyzed the performance of LT codes using different metrics like average degree, release probability and overhead.

Summary of the Issues in the Existing System

Based on the above literature review, we find that the following are the issues that need to be resolved in the existing Conventional Degree Distribution (CDD) functions designed for LT codes in order to achieve the optimal performance of the communication system:

- Reduce transmission of redundant degree one encoded symbols
- Premature termination of the LT decoder
- Minimize encoder/decoder delay
- Minimize encoder/decoder overhead
- Maintaining the constant ripple factor
- Effective utilization of the bandwidth

Problem Description

To address the above issues, our proposed work emphasizes on understanding the whole LT encoding process as a bipartite structure and to design a joint degree distribution of vertices involved in the graph as an interesting research for finding an optimal degree distribution solution for LT codes. From the research work of (Stanton and Pinar, 2012), we find that it is possible to generate random instances of graphs with the same joint degree distribution for LT encoding process where the joint degree matrix (or JDM for short) model which is a version of JDD for a network structure. In essence, the JDD gives (for each i and j) the probability that an edge of the graph connects a vertex of degree i to a vertex of degree j, while JDM tells us the exact number of edges between vertices of degrees i and j.

In our proposed work, we have designed a restrictive model adapted from the above Joint Degree Distribution (JDD) model (Stanton and Pinar, 2012; Czabarka *et al*., 2013), to maximize bandwidth utilization and also to resolve the above issues. In our work, we have given a simplified proof of the necessary and sufficient conditions for a matrix to be a joint degree matrix for realizing LT encoding as a bipartite graph, which includes a general method for constructing realization of joint degree distribution. Based on the proposed JDD based LT codes, we investigate the performance of the LT decoder for small number of input symbols without sending the excessive encoded symbols for recovery. That is, the number of encoded symbols that are needed for recovery is well-defined. In this model, only k encoded symbols are sufficient to successfully decode k source symbols without compromising the complexity of LT codes.

Proposed Work

In LT codes, the source symbols are chosen randomly, which may be redundant. There are chances that the same source symbol may be repeatedly chosen for encoding (random sampling with replacement). In the case of degree 1 encoded symbol, redundant choice of source symbols may reduce the performance of LT decoding.

In our initial design for distribution, the effects of various degree combinations in encoded symbols were studied. The best performance in terms of throughput, bit error rate, encoder/decoder delay and overhead was obtained using degree 1 and degree 2 combinations of encoded symbols.

The focus of the model is in restricting the degree 1 encoded symbols. We introduce a variant of degree distribution traditionally used in LT codes which permits only degree 1 and degree 2 encoded symbols. Here, we have restricted the choice of source symbol for contributing to degree 1 encoded symbol to be random but distinct. If a symbol is chosen redundantly, the choice is discarded and repeated until another distinct symbol is chosen (random sampling without replacement). This sometimes leads to an uncoded state (i.e., all encoded symbols are of degree 1), when all encoded symbols are of degree 1 and equal or greater than the number of source symbols. This state also reduces performance of recovery, if any encoded symbols are lost during transmission. The objective of this design is to find the optimal number of degree 1 encoded symbols which provides best throughput and low BER. We extended this idea to JDD to improve the bandwidth utilization, as the conventional LT codes using the existing degree distributions achieves the maximum throughput only by consuming more bandwidth.

Design of Joint Degree Distribution (JDD)

In this section, we introduce our design of LT encoder using joint degree distribution (JDD). The degree distribution employed in the encoding process of our proposed design is analogous to a balanced irregular bipartite graph G (U,V,E). U and V are subsets of vertices of G and have the same cardinality (thus making the bipartite graph balanced) and E is the set of edges from U to V. The degree of vertices are different in U and V and hence G is irregular bipartite but balanced. U represents left nodes also called as source symbols and V represents right nodes also called as encoded symbols obtained by performing a XOR operation on two or more source symbols (U). Edges E run between vertices in U (source symbols) to vertices in V (encoded symbols from corresponding source symbols in U). The system model for LT code using modified degree distribution, JDD is shown in Fig. 3.

Linking Probability Matrix (LPM)

Lemma 1

For JDD, the linking probability matrix LPM can be modeled as $d_{max} \times d_{max}$ sized matrix, which describes what is the probability of every degree i node u_i in U linking with every degree j node v_j in V, in such a way that the sum of linking probabilities for every degree i node u_i is always equal to 1. In the

design of JDD, the probability of degree 1 nodes u_1 in U linking with degree 1 nodes v_1 in V is always zero as given in Equation 1:

$$LPM = \begin{pmatrix} L_{1,1} & \cdots & L_{1,d_{\max}} \\ \vdots & \ddots & \vdots \\ L_{d_{\max},1} & \cdots & L_{d_{\max},d_{\max}} \end{pmatrix} \quad (1)$$

Joint Degree Matrix (JDM)

Generally, the Degree Distribution (DD), $\rho(d)$ is the probability of a random encoded symbol $v \in V$, to have degree d, $1 \leq d \leq d_{max}$. The Joint Degree Distribution (JDD), $\rho(d,r)$ is the probability of a random encoded symbol v € V of degree d to have an edge with a random source symbol $u \in U$ of degree r. The degree vector D also called as degree sequence is the count of the number of nodes of degree k, ($1 \leq k \leq d_{max}$) in the given graph.

The Joint Degree Matrix (JDM) is a matrix of size $d_{max} \times d_{max}$, where d_{max} is the maximum degree of a node in the graph G. Each element J_{ij} represents the count of edges that run between nodes of degree i to degree j (from U to V). The degree vector can be obtained from the JDM using the Equation 2:

$$JDM = \begin{pmatrix} J_{1,1} & \cdots & J_{1,d_{\max}} \\ \vdots & \ddots & \vdots \\ J_{d_{\max},1} & \cdots & J_{d_{\max},d_{\max}} \end{pmatrix} \quad (2)$$

$$J_{i,j} = \frac{\left(i \times \left(n \times L_{i,j}\right)\right)}{d_{\max}} \quad (3)$$

where, each element $J_{i,\ j}$ of JDM can be found by using Equation 3.

From Equation 1-3, the degree sequence and the corresponding total number of edges present in the entire graph are determined for k, r = 1,2, ….., d_{max} as given in Equation 4:

$$D_k = \frac{1}{k}\left(J_{k,k} + \sum_{\substack{r \\ for\, r \neq k}}^{d_{\max}} J_{k,r}\right) \quad (4)$$

If the number of nodes in both U and V is n and the linking probability is p, then the network will have an average nodal degree as Equation 5:

$$d_{avg} = np \quad (5)$$

The linking probability p can be determined by using a well defined JDD, which examines each pair of connected nodes and represents their respective nodal degrees in JDM.

Node Perspective Degree Distribution

This degree distribution with respect to nodes in U and V can be inspected as follows:

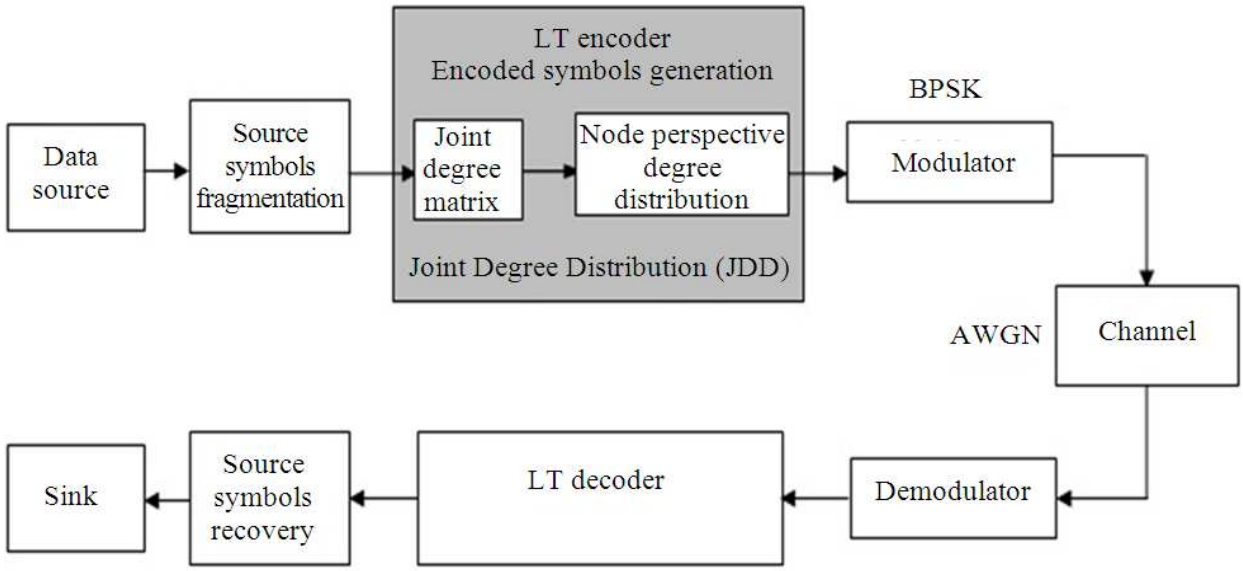

Fig. 3. JDD based LT codes system model

Determining Degree One Nodes

The variable degree distribution which is considered in the proposed design does not have edges from any node with degree 1 in *U* to a node with degree 1 in *V*. There is no edge connecting only one source symbol contributing to one encoded symbol. For successful recovery of source symbols, it is ensured that a source symbol contributes in the encoding process of more than an encoded symbol or more than one source symbols are XORed to form an encoded symbol.

Lemma 2

For the given bipartite graph *G* of d_{max} as 2, with known JDM and derived D_k, the linking probability of the number of degree one nodes in *V* connecting with degree one nodes in *U* is zero and the linking probability of the number of degree one nodes in *V* linked with degree one node in *U* is one.

Lemma 3

For the given bipartite graph *G* of d_{max} as 2, with known JDM and derived D_k, the linking probability of the number of degree two nodes in *V* connecting with degree one nodes in *U* is same as that of connecting with degree two nodes in *U* is ½.

Proof

Given the graph *G* with specific JDD and the cardinality n of both *U* and *V* as 10 with the maximum degree d_{max} as 2. By applying Lemma 1, the LPM and JDM are given in Equation 6 and 7:

$$JDM = \begin{pmatrix} J_{1,1} & J_{1,2} \\ J_{2,1} & J_{2,2} \end{pmatrix} = \begin{pmatrix} 0 & 5 \\ 5 & 5 \end{pmatrix} \tag{6}$$

$$LMP = \begin{pmatrix} L_{1,1} & L_{1,2} \\ L_{2,1} & L_{2,2} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1/2 & 1/2 \end{pmatrix} \tag{7}$$

For $k = 1$, $d_{max} = 2$, we get D_1, the number of nodes in V with degree 1 are found in Equation 8 and 9:

$$D_1 = \frac{1}{1}\left(J_{1,1} + \sum_{\substack{r=1 \\ for\, r \neq k}}^{2} J_{1,r} \right) \tag{8}$$

$$= \left(J_{1,1} + J_{1,2} \right) \tag{9}$$

By using the above JDM of the graph *G* given in Equation 7 with derived degree sequence D_1 from Equation 8, first we randomly choose both D_1 degree one *V* nodes and D_1 degree one *U* nodes of the bipartite graph as shown in (Fig. 4a).

Determining Degree Two Nodes

In the same way, for $k = 2$, $d_{max} = 2$, we get D_2, the number of nodes in *V* with degree 2 are found as given in Equation 10 and 11:

$$D_2 = \frac{1}{2}\left(J_{2,2} + \sum_{\substack{r=1 \\ for\, r \neq k}}^{2} J_{2,r} \right) \tag{10}$$

$$= \frac{1}{2}\left(J_{2,2} + J_{2,1} \right) \tag{11}$$

By using the above JDM of the graph *G* given in Equation 7 with derived degree sequence D_2 from Equation 10, we choose both D_2 degree two *V* nodes and D_2 degree two *U* nodes of the given graph.

Determining the Edges for Degree One Nodes in V

The total number of edges E_{Total} of the given graph can be calculated using the degree sequence D_k of the graph as Equation 12:

$$E_{Total} = \sum_{k=1}^{d_{\max}} kD_k \tag{12}$$

But, for the graph *G* with n nodes of degree d_{max}, the maximum number of possible edges in G is bounded as in Equation 13:

$$E_{\max} = nd_{\max} \tag{13}$$

The average nodal degree d_{avg} can be found as Equation 14:

$$d_{avg} = \frac{1}{n}\sum_{k=1}^{d_{\max}} kD_k \tag{14}$$

According to Lemma 2, there exists no edge between degree one nodes in *V* and degree one nodes in *U*, because $J_{1,1} = 0$. Hence the proposed distribution randomly selects a distinct degree two node from *U* to generate a corresponding encoded symbol i.e., a node with degree one in *V*.

Determining the Edges for Degree Two Right Nodes

The major contribution of our design lies here. A degree two node $n_v \in V$ has edges from two nodes n_{u1}, $n_{u2} \in U$ with degree k_1 and k_2 respectively such that $k_1 \neq k_2$, $1 \leq (k_1, k_2) \leq d_{max}$.

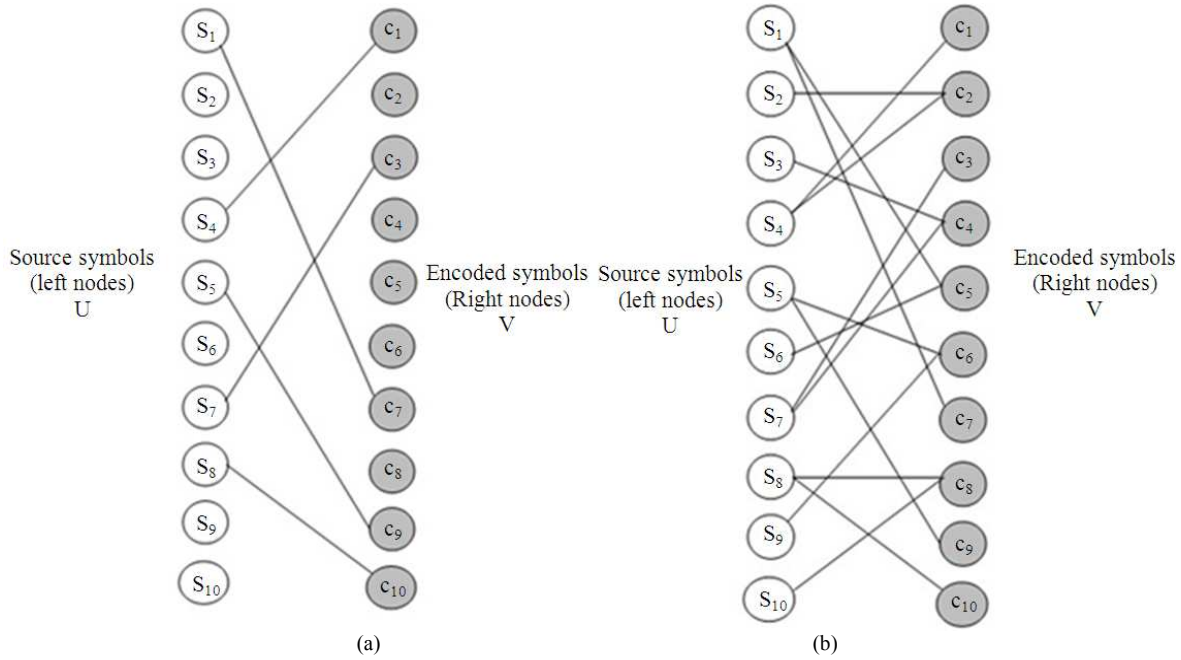

Fig. 4a and b. The bipartite graph representation of JDD

i.e., every degree two nodes in V has one edge with degree one node in U and another edge with degree two node in U such a way that the nodal degree of the nodes in U does not exceed d_{max}.

The complete, balanced and irregular bipartite graph for the given JDD is shown in (Fig. 4b). This distribution when applied in the LT encoder facilitates successful termination of the LT decoder, which enhances the throughput performance with low encoding/decoding delays.

In any transmission, the successful decoding of source symbols purely relies on the degree distribution used in the encoder. The decoding process will succeed only if there is at least one received encoded symbol in the ripple at every intermediate stage until the entire source symbols are recovered. When all the source symbols have been recovered and with no encoded symbol in the ripple, then the decoder terminates successfully. If the ripple gets emptied during any of the intermediate stages, then the decoder prematurely terminates with source symbols pending to be recovered.

Thus, it is evident that the ripple size has to be maintained ≥1, until all source symbols are recovered. The constraint on ripple size ≥1 is satisfied in the proposed JDD distribution for encoding the symbols. The proposed JDD based LT encoder ensures the high throughput performance of the LT decoder. The joint degree distribution which is the base of our work ensures that all source symbols are included in the encoding process, thereby maintaining the ripple size at the decoder constant with degree one symbols. The LT encoder using our proposed distribution runs in linear time, since the average degree d_{avg} of the encoded symbol is independent of the symbol length as in traditional LT codes.

Simulation Results

Simulation Environment

The JDD and CDD based LT codes have been implemented in 'C' language. The source data is generated using random binary data generator. The input data of size 10^6 bits is fragmented in to 100 source symbols of symbol length 10,000 bits. The performance of LT codes using CDD and proposed JDD are studied for transmission over AWGN channel using Binary Phase Shift Keying (BSPK) as the modulation scheme.

Performance Metrics

The performance of JDD over conventional system for various aspects like delay, throughput, constant ripple size and bit error rate performance are discussed below.

Delay in Encoding/Decoding Process

The observations for the average encoder/decoder delay and the source symbols recovered are tabulated as shown in Table 1 and 2.

Figure 5 illustrates the average encoder/decoder delay performance of CDD for varying the encoded symbols size.

Decreasing Ripple Size

The ripple is a buffer used in decoding to store the degree 1 encoded symbols. This ripple should be maintained with at least one degree 1 symbol to continue decoding until successful recovery of all source symbols. Figure 6 describes the decreasing ripple size of CDD and JDD.

Throughput

Figure 7 describes the throughput performance of CDD and JDD for varying the number of encoded symbols.

BER Vs SNR

The BER Vs SNR performance comparison between CDD and JDD based LT codes over AWGN channel using BPSK modulation is analyzed for the varying number of degree 1 nodes from 10 to 50 with an increment of 10 for the varying SNR from 0dB to 8.5dB in increments of 0.5dB. The performance with number of degree 1 nodes as 50 is shown in Fig. 8.

Discussion

The objective of this performance study is to control the premature termination of the decoder in LT codes. Our study about the factors that are significant for successful decoding revealed that, the degree distribution of symbols is sensitive to recovery of source symbols. In order to understand the role of degree distribution in LT codes, the various sets of encoded symbols in combinations of {degree 1, degree 2}, {degree 1, degree 3}, {degree 1, degree 4} and etc have been implemented. It was observed that, the encoded symbols of combination (degree 1, degree 2} performed much better due to least overhead ($\varepsilon = 2$) compared to other combinations. This degree distribution also conserves bandwidth as only 300 encoded symbols need to be transmitted for 100 source symbols. The other two degree distribution combinations require the size of encoded symbols to be up to 8 to 12 times the size of source symbols.

Motivated by the above simulation results, a Joint Degree Distribution (JDD) based LT codes have been implemented with same encoded symbols of combination (degree 1, degree 2} to increase reliability and success in recovery of all source symbols. In the JDD design, the role of degree 1 encoded symbol is further restricted by not allowing a degree 1 encoded symbol to be connected to a degree 1 source symbol. There are no degree 1 source symbols. A source symbol may contribute to more than one degree 1 encoded symbols or any combination of degree 1, degree 2 encoded symbols, thereby eliminating the chances of an uncoded state in contrast to the conventional system. This increases the reliability of the design as even if there is a loss in the encoded symbols received, the source symbols can be recovered from their other contributions.

From the values tabulated in Table 1 for a transmission of 100 encoded symbols, it can be observed that the encoder/decoder delay is relatively same for JDD and CDD based LT codes, but JDD is better in terms of the number of source symbols successfully recovered.

To match this performance in conventional system, approximately 300 encoded symbols have to be transmitted and the encoding/decoding delay increases correspondingly as shown in Table 2.

Figure 5 demonstrates that the average encoder/decoder delay of CDD linearly increases with respect to the varying number of encoded symbols. But, using JDD, almost approximately the same average encoder/decoder delay could be achieved with maximum throughput. From Table 1 and 2 and Fig. 5, it can be concluded that JDD outperforms CDD in terms of delay.

Figure 6 explains the constantly decreasing ripple for CDD and JDD. In CDD, the count of the symbols in the ripple was continuously monitored to find that, it started with a randomly high value and dropped to zero resulting in premature termination of the decoder for the smaller number of encoded symbols size like 100, 150. The size of the ripple also varied for different trials.

The same observation for JDD started with exactly the number of degree 1 encoded symbols allowed for that transmission (say 10 or 20) and got decremented for subsequent iteration. When it reached zero, all symbols were decoded resulting in successful termination of the decoder. The initial ripple size and its decrements remained constant and independent of the trials. Thus, JDD makes use of the ripple optimally and effectively compared to CDD as shown in Fig. 6.

CDD with 10% degree 1 encoded symbols, the better throughput performance (that obtained with degree 1 encoded symbol) is almost reached. This reduction in number of degree 1 encoded symbols obviously increases throughput (average of trails tabulated in Table 2). Whereas in JDD, since redundancy is completely eliminated and all degree 1 encoded symbols are derived from distinct source symbols and since inclusion of all symbols is ensured the maximum performance is ideal 100% (average of trails tabulated in Table 1). Thus, JDD gives maximum throughput without any increase in bandwidth as shown in Fig. 7.

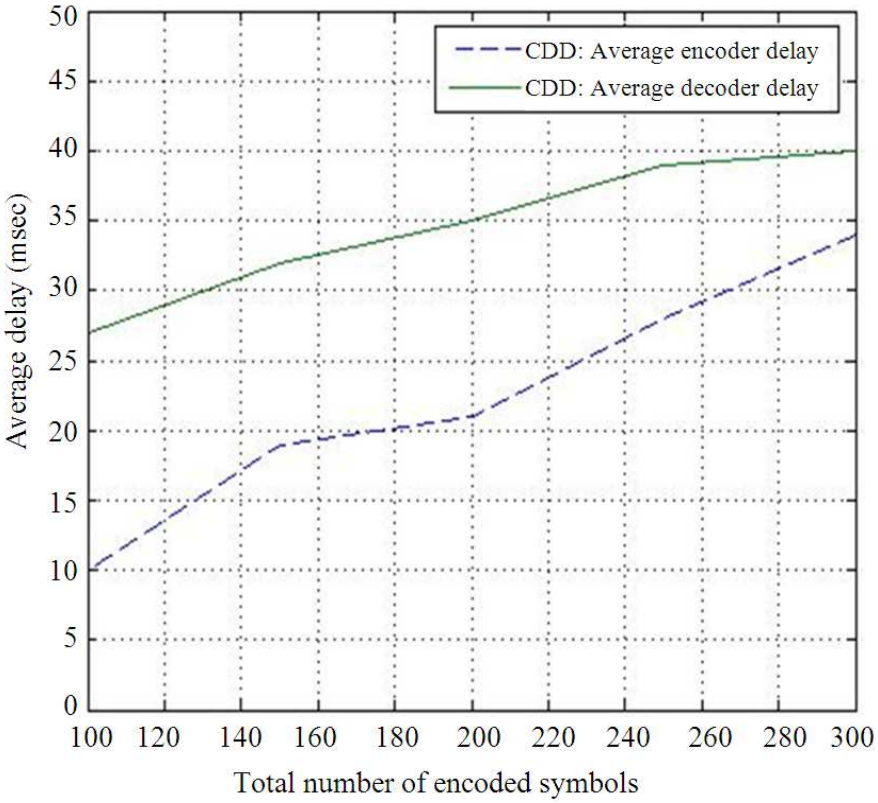

Fig. 5. Average encoder/decoder delay performance of CDD

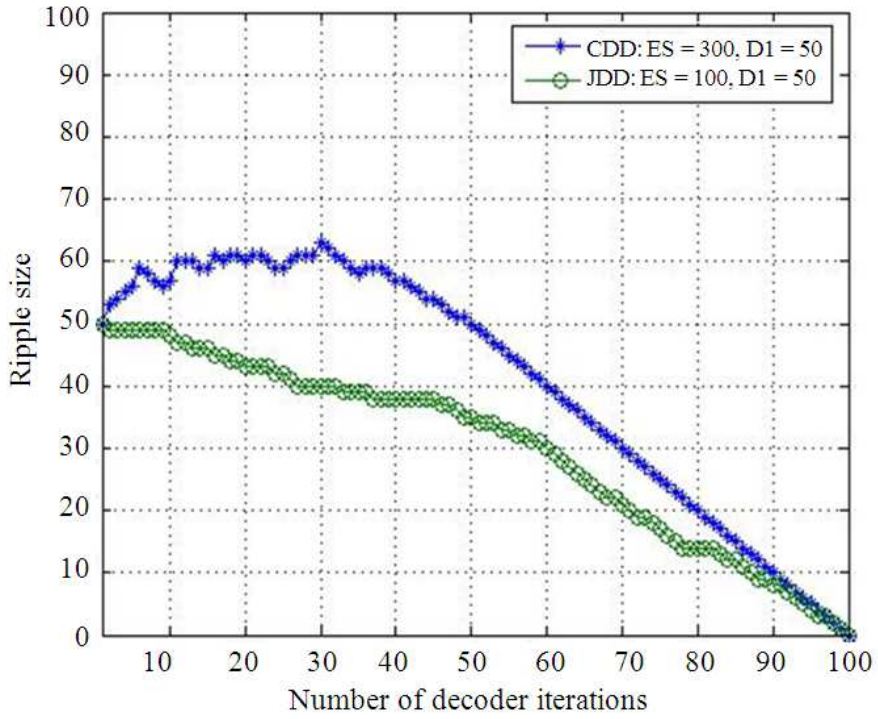

Fig. 6. Decreasing ripple size of CDD and JDD

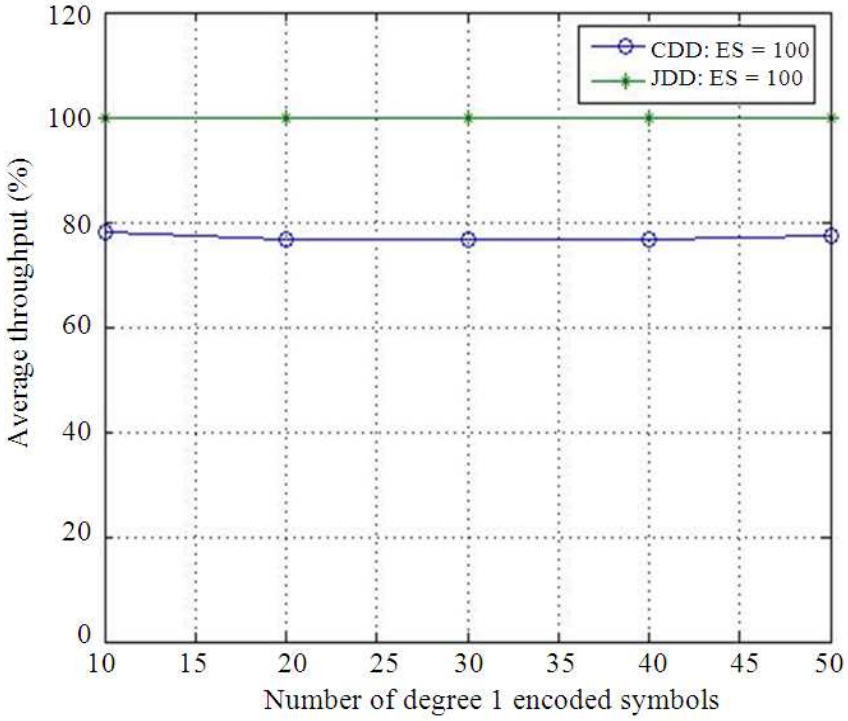

Fig. 7. Comparative throughput performance of CDD Vs JDD

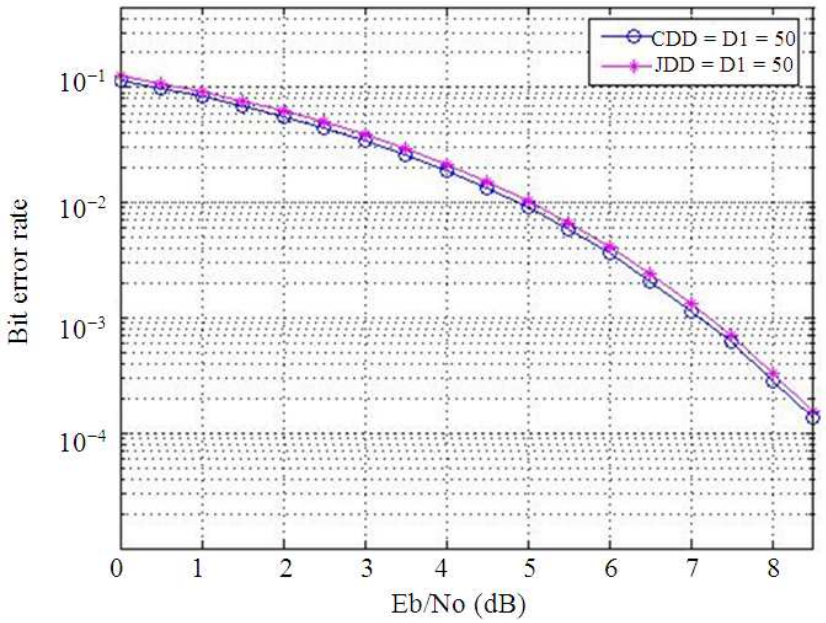

Fig. 8. BER Vs SNR performance of CDD (ES = 300) and JDD (ES = 100) for D1 = 50

Table 1. Delay performance of CDD Vs JDD for encoded symbols size ES of 100

Degree distribution	Average encoding delay (ms)	Average decoding (ms) delay	Number of encoded symbols transmitted	Average no. of successfully recovered source symbols
CDD	10	27	100	82
JDD	11	32	100	100

Table 2. Performance of CDD based LT code for varying number of encoded symbols

Number of encoded symbols	Average encoding delay (msec)	Average decoding delay (msec)	Average no. of successfully recovered source symbols
100	10	27	82
150	19	32	94
200	21	35	98
250	28	39	99
300	34	40	100

In CDD, since the selection of source symbols to be encoded allows redundancy, the maximum throughput performance can be achieved by consuming 3 times of the bandwidth.

From the results obtained in Fig. 8, it was observed that JDD based LT codes achieves almost the same BER with less bandwidth and reaches maximum throughput compared to CDD in which a higher bandwidth is required to maximize the throughput for the same BER. This results in increase in cost of the encoding/decoding overhead ($\varepsilon = 3$) and the encoding/decoding delay in CDD. This is the major advantage of JDD over CDD.

From the above discussions on the simulation results, it is inferred that the careful design of JDD based LT codes outperforms the conventional systems over AWGN channel in terms of throughput, encoding/decoding overhead, delay and also maintains a constantly decreasing ripple for successful decoding. In addition, JDD ensures that the efficient utilization of the bandwidth can be achieved by using only 'k' encoded symbols for 'k' source symbols, unlike in conventional systems (Zhiliang *et al*., 2012).

Conclusion

In this study, a modified version of degree distribution, the Joint Degree Distribution (JDD) is proposed. JDD optimizes the bandwidth utilization for same BER performance compared to traditional degree distribution schemes used in LT codes. JDD also proves itself to be better in terms of maximizing throughput, minimizing encoding/decoding delay and overhead and makes effective use of the ripple. Limiting the number of degree 1 encoded symbols improves the reliability of the transmission. JDD seems to be more useful in wireless transmissions where efficient utilization of bandwidth is essential. These advantages contribute to validity of using JDD based LT codes as a promising coding strategy in real time data transfer. The limitations of this study are that JDD has been designed only for the combinations of degree 1 and degree 2 encoding symbols and also its performance has been tested in AWGN channel. But, the results of this study give motivation to extend JDD for analyzing the performance of LT codes over fading channel conditions and also aims to propose a modified JDD involving higher degree encoded symbols as the future research perspective.

Funding Information

The authors have no support or funding to report.

Author's Contributions

All authors equally contributed in this work.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

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

Czabarka, E., A. Dutle, P. Erdös and I. Miklós, 2013. On realizations of a joint degree matrix. arXiv: 1302.3548.

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

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

Lu, F., C.H. Foh, J. Cai and L.T. Chia, 2009. LT codes decoding: Design and analysis. Proceedings of IEEE International Symposium on Information Theory, Jun. 28 2009-Jul. 3, IEEE Xplore Press, Seoul, South Korea, pp: 2492-2496. DOI: 10.1109/ISIT.2009.5206067

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

Palanki, R. and J.S. Yedidia, 2004. Rateless codes on noisy channels. Proceedings of the International Symposium on Information Theory, Jun. 27-Jul. 2, IEEE Xplore Press, Chicago, IL, USA, pp: 37-37. DOI: 10.1109/ISIT.2004.1365075

Salkuyeh, M.A., F. Hendessi and T.A. Gulliver, 2013. Data dissemination with rateless coding in a grid vehicular topology. Eur. J. Wireless Commun. Netw., 2013: 106-106. DOI: 10.1186/1687-1499-2013-106

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

Stanton, I. and A. Pinar, 2012. Constructing and sampling graphs with a prescribed joint degree distribution. ACM J. Experimental Algorithms. DOI: 10.1145/2133803.2330086

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

Zao, J.K.., M. Hornansky and P. Diao, 2012. Design of optimal short-length LT codes using evolution strategies. Proceedings of IEEE Congress on Evolutionary Computation, Jun. 10-15, IEEE Xplore Press, Brisbane, QLD, pp: 1-9. DOI: 10.1109/CEC.2012.6256147

Zhiliang, Z., L. Sha, Z. Jiawei, Z. Yuli and Y. Hai, 2012. Performance analysis of LT codes with different degree distribution. Proceedings of 5th 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.P., G.X. Zhang and Z.D. Xie, 2009. Suboptimal degree distribution algorithm of LT codes of digital fountain. J. Applied Sci. Electron. Inform. Eng., 27: 6-11.