IJRSET JUNE 2017 Volume 4, Issue 6 Pages: 1-5
ISSN 2394-739X
International Journal for Research in Science Engineering and Technology

PERFORMANCE REVIEW OF FOUNTAIN CODES FOR NOISY CHANNELS

[1] Gurpreet Kour,
[1] Student,Department of Electronics and Communication
[1] Vaishno College of Engineering and Technology, H.P, India

ABSTRACT: Current networks use unicast-based protocols such as the transport control protocol (TCP), which require a transmitter which continually send the same packet to the reciever until acknowledged by the receiver. It can be seen that this architecture does not goes well when more than one user access a server concurrently and is extremely inefficient when the information transmitted is always the same. An alternative approach for this where packets are not ordered and the recovery of some subset of packets will allow for successful decoding. This class of codes, called fountain codes, has greatly influenced the design of codes for binary erasure channels (BECs), a well-established model for the Internet. Luby transform codes (LT codes) are the first class of practical fountain codes that are near-optimal erasure correcting codes. The LT codes employs a particular simple algorithm based on the exclusive or operation to encode and decode the message. This paper will review Fountain Codes and its practical applications i.e LT codes and Raptor codes.

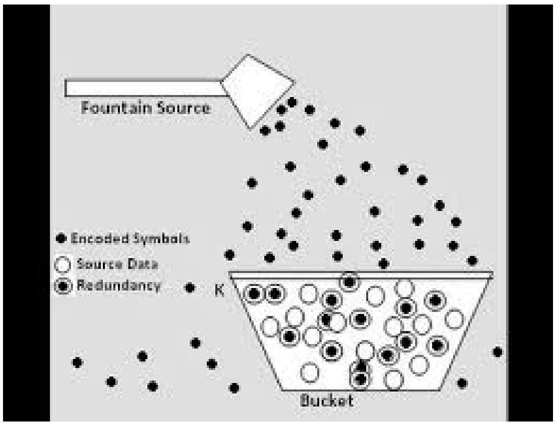
Keywords: [fountain codes, Rateless codes, degree distribution, LT codes, Raptor codes.]

1. INTRODUCTION

Data is transmitted in the form of packets, on the Internet. Each packet contain a header that shows the source address and the destination address of the packet.These packets are routed on the network from the sender to the receiver. Due to various channel complexities, some packets may get lost and never reach their destination. Reliable transmission of data over the Internet is always desirable. Current networks use unicast-based protocols such as the transport control protocol (TCP/IP) which ensures reliability by retransmitting packets within a transmission window whose reception has not been acknowledged by the receiver. It is well known that such protocols exhibit poor behavior in many cases, such as transmission of data from one server to multiple receivers, or transmission of data over heavily impaired channels, such as poor wireless or satellite links. Moreover, ack-based protocols such as TCP perform poorly when the distance between the sender and the receiver is long.[1].For these reasons, other transmission solutions have been proposed. One of such solutions is based on coding. The original data is encoded using some linear erasure correcting code.If during the transmission some part of the data is lost, then it is possible to recover the lost data using erasure correcting algorithms. In effect, TCP and other unicast protocols place strong importance on the ordering of packets to simplify coding at the expense of increased traffic. An alternative approach is where packets are not ordered and the recovery of some subset of packets will allow for successful decoding. This class of such codes, called fountain codes, was pioneered by a startup called Digital Fountain and has greatly influenced the design of codes for binary erasure channels (BECs), a well-established model for the Internet. [1][5]

2. FOUNTAIN CODES

Fountain codes, are a class of erasure correcting codes. These codes can produce an unlimited flow of encoding data blocks, i.e., they are rate-less. In this class of codes, input and output symbols can be binary vectors of arbitrary length. Each output symbol is the sum of a randomly and independently chosen subset of the input symbols. i.e each output symbol is the addition of some of the input. The digital fountain was devised as the ideal protocol for transmission of a single file to many users who may have different access times and channel fidelity. The output packets of digital fountains has properties similar to a fountain of water, when you fill your cup from the fountain, you do not care what drops of water fall in, but your cup fills enough to quench your thirst. Similarly with digital fountain, a client obtains encoded packets from one or more servers, and once enough packets are obtained, the client Can reconstruct the original file,which packets are obtained should not matter.



Fountain codes are ideally more suited for transmitting information over computer networks. A server sending data to many recipients can implement a Fountain code for a given piece of data to generate an infinite stream of packets. As soon as a receiver requests data, the packets are copied and forwarded to the recipient. The recipient collects the output symbols, and leaves the transmission, it then uses the decoding algorithm to recover the original symbols.

In this paper, we will not address these and other applications, but will instead focus on the theory of such codes. In order to make Fountain codes work in practice, one needs to ensure that they possess a fast encoder and decoder, and that the decoder is capable of recovering the original symbols from any set of output symbols whose size is close to optimal with high probability. We call such Fountain codes universal. The first class of such universal Fountain codes was invented by Luby and is called LT-codes.

3. LT CODES

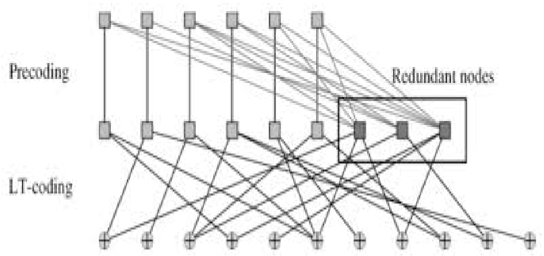
LT codes are the first practical rateless codes for the binary erasure channel. The encoder can generate as many encoding symbols as required to decode k information symbols. The encoding and decoding algorithms of LT codes are simple; they are similar to parity-check processes. LT codes are efficient in the sense that the transmitter does not require an acknowledgement (ACK) from the receiver. This property is desired in multicast channels because it will significantly decrease the overhead incurred by processing the ACKs from multiple receivers. [4]

LT ENCODING: The encoding process starts by dividing the uncoded message into N blocks of equal length. Encoded packets are then produced with the help of a random number generator. The degree n is considered as the random number with the range, $1 \leq n \leq N$, and the next packet is chosen at random. Exactly n blocks from the source message are randomly chosen. If Mi is the ith block of the message, the data partition of the packet is computed as follows:

$Mi1 \oplus Mi2 \oplus \cdots \oplus Min$

Where {i1, i2,…,in} are the randomly chosen keys for the n blocks included in this packet. A prefix is appended to the encoded packet, that define the total blocks as N in the source message, determine n blocks have been exclusive-ored into the data segment of this packet, and the list of keys {i1, i2, …, in}. Finally, some of the error-detecting code is applied to the packets to determine error, and then only that packet is transmitted. This process continues until the receiver send signals to the sender that the message has been received and it is successfully decoded.

LT DECODING: The decoding process uses the xor operation used by the sender to retrieve the encoded message. If the currently received packet isn't pure, or if it replicates a packet that has already been processed, the current packet is



discarded. If the current cleanly received packet is of degree $n > 1$, it is first processed with all fully decoded blocks in the message queuing area, and then stored in a buffer area if its degree is greater than 1. Whenever a clean packet of degree $n = 1$ is received, it is moved to the message queueing area, and then it is matched against all the packets of degree $n > 1$ residing in its buffer. It is xored with the data portion of any buffered packet that was encoded using the block Mi, the degree of that matched packet is decremented, and the list of keys for that packet is adjusted to reflect the application of Mi. When this process unlocks a block of degree $n = 2$ in the buffer, that block is deducted to degree 1 and is in need to move on to the message queueing area, and then processed against the packets remaining in the buffer area. When all N blocks of the data packets have been directed to the message queueing area, the receiver signals the sender that the data packets has been successfully decoded. This decoding procedure works because $A \oplus A = 0$ for any string A. After $n - 1$ divided blocks have been exclusive-ored into a packet of degree n, the original encoded content of the mismatched block is all that remains as same.

With the encoding symbols and the indices of their neighbors, the decoder can recover information symbols with the following three-step process, which is called LT process:

1) All encoding symbols of degree one, i.e., those which are connected to one information symbol, are released to cover their unique neighbor.

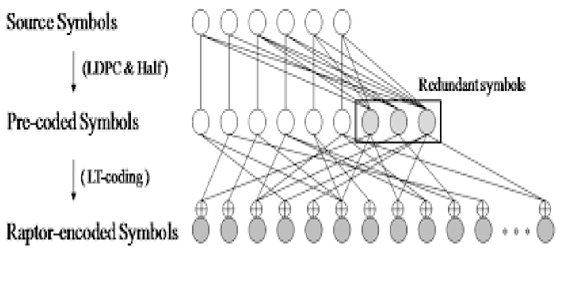
2) The released encoding symbols cover their unique neighbor information symbols. In this step, the covered but not processed input symbols are sent to ripple, which is a set of covered unprocessed information symbols gathered through the previous iterations.

3) One information symbol in the ripple is chosen to be processed: the edges connecting the information symbol to its neighbor encoding symbols are removed and the value of each encoding symbol changes according to the information symbol. The processed information symbol is removed from the ripple. The analysis of LT codes is based on the analysis of LT processes, which is the decoding algorithm.The importance of having a good degree distribution of encoding symbols arises from this analysis. As a result, the Robust Soliton distribution is introduced as the degree distribution.

4. RAPTOR CODES

In practice, the Fountain code with Belief Propagation decoding algorithm fails sometimes even for large block sizes since due to the random selection of input nodes some of the input nodes are not covered.

The solution for this is a Low Density Parity Check (LDPC) precoder, which is called Raptor code. In practice, the bipartite graph of LDPC and Fountain code are combined so that the decoding complexity is not increased. The main idea behind Luby-Transform (LT) codes and Raptor codes is to design the degree distribution (such as the robust Soliton distribution, or one of its variant) of a coded packet. The degree of a coded packet indicates the number of input packets used to generate the coded packet. LT codes and Raptor codes are irregular codes. LT coding is done in two steps, first the encoder randomly selects the degree, whose expected probability is dictated by the degree distribution. In the next step the encoder, randomly selects input packets, the number of input packets selected is given by the degree selected in the first step, and perform XOR addition of those input packets. Decoding is performed using back-substitution. The decoder looks for those coded packet with one unknown input packets, and decode the unknown packet. It then substitutes this decoded packets in all the other coded packets which had been generated using this decoded packet as one of its constituent encoding packet. The decoder continues to repeat this process of decoding and substitution until it has not decoded all the k input packets. If it is unable to decode k input packets, then it requests for additional coded packets to be transmitted by the server. Raptor (Rapid Tornado) codes is a special class of LT codes. The design of Raptor codes



is motivated by the fact that due to the random nature of selecting the input packets, there is always a non-zero probability that some of the input packets may never be selected for coding in LT codes. To address this problem, in Raptor code the input symbols are first precoded, and then LT coding procedure takes place. The redundant packets can be generated by randomly coding the input packets using XOR addition. After the precoding the output packets are generated using LT coding, whose input packets are given by the concatenation (c1, . . . , ck, y1, . . . , yj). Decoding Raptor codes is done using inactivation decoding.

5. BARRIERS TO ADOPT DIGITAL FOUNTAIN CODES

Decoding delay of erasure codes, pollution attack, and the index coding problem. While the use of erasure code improves the bandwidth performance of a broadcast network, it has a disadvantage of incurring a decoding delay. For example, for a client who has packet c1 and wants packets c2 and c3, coded packets c2 $\oplus$c3 and c1 $\oplus$ c2 are both linearly independent with respect to c1, however only the latter coded packet can be instantly decoded by the client. Another problem related to security aspects of erasure codes is that of pollution attack. If the client admits even a single malicious coded packet from a malicious user, then during the decoding process, all the decoded packets will be corrupted.

6. APPLICATIONS:

Erasure codes have been proposed as an efficient remedy to improve the reliability and scalability of data transmission over erasure channels. In erasure coding, the coded packets are generated by linearly mapping the packets with probability, so these codes have high applications in multicasting and broad casting.another application of fountain codes is data storage, Packet erasure is one the fundamental and inevitable characteristic in data transmission and data storage system. For example routers may drop a packet due to congestion. Similarly a file in a data storage system can be erased due to component failures. The problem of packet erasure exuberates for data transmission on wireless channel due to the shared medium of transmission resulting in packet collisions. In addition to packet collision, for wireless channel, packet may also be erased due to channel fading, additive white Gaussian noise (AWGN) and signal attenuation. Traditional approach of dealing with packet erasure is to use replication and retransmission. The method of replication and retransmission introduces control overhead. For data storage system, replication provides limited reliability. For instance in the event that the original file and the replicated files are both erased, then the data storage system can not recover the original file. Similarly the use of retransmission technique for data transmission system is dependent on packet acknowledgement control frame from the client. It is also possible that an acknowledgement frame can also be erased due to the same reasons as the original data packet, erroneously resulting in the retransmission of those data packets which the client has already received. For wireless networks, the transmission of acknowledgement frame occupies the wireless channel medium and therefore adversely affects the transmission bandwidth. [2]

CONCLUSION

Traditional approaches to deal with system erasure are to use retransmission and replication techniques, which limits the reliability of the system, and adversely affects the throughput performance of the system. A series of Fountain codes - Tornado codes, LT codes, and Raptor codes - have been proposed, and patented, to address the decoding complexity.Thus the Fountain Codes are a new class of codes designed for robust, resynchronized, and scalable transmission of information from multiple senders to multiple receivers in a reliable manner over computernetworks. The hypothesis of Fountain Codes is very exciting, and also provides new imminent into the theory of parity check codes. New asynchronous multicast applications using Fountain Codes is utilized by software simulation and hardware implementation.

REFERENCES

[1] Amin Shokrollahi," Raptor Codes" IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 52, NO. 6, JUNE 2006 2551.

[2] D.J.C. MacKay," Fountain codes",CAPACITY APPROACHING CODES DESIGN AND IMPLEMENTATION
SPECIAL SECTION IEE Proceedings online no. 20050237 doi:10.1049/ip-com:20050237 Paper received 23rd May 2005.

[3] Zhong Cheng, Jeff Castura, Yongyi Mao, "On the Design of Raptor Codes Binary-Input Gaussian Channels", Proc. IEEE International Symposium
on Information Theory, Nice, France, June 2007.

[4] M.Luby, "LT-codes",Proc. 43rd Annual IEEE Symposium on Foundations of Computer Science(FOCS), Vancouver, BC, Canada, Nov. 2002

[5] "LT-codes," in Proc. 43rd Annu. IEEE Symp. Foundations of Computer Science (FOCS), Vancouver, BC, Canada, Nov. 2002, pp. 271–280.

[6] A. Shokrollahi and M. Luby, "Systematic Encoding and Decoding of Chain Reaction Codes," U.S. Patent 6,909,383, Jun. 21, 2005.