Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 23, 2019

S. Yang
X. Huang
CUHK(SZ)
R.W. Yeung
CUHK
J. Zao
NCTU

October 20, 2018

BATS Code Scheme for Multi-hop File Delivery
draft-yang-nwcrg-bats-code-00

Abstract

This document describe a BATS code scheme for communication through a multi-hop network. BATS code is a class of efficient linear network coding scheme with a matrix generalization of fountain codes as the outer code, and batch-based linear network coding as the inner code.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 23, 2019.

Table of Contents

1. Introduction

This document specifies a BATS code [BATS] scheme for file delivery applications in multi-hop networks. The BATS code described here includes an outer code and an inner code. The outer code is a matrix generalization of the fountain codes (see also the RapterQ code described in RFC 6330 [RFC6330]), which inherits the advantages of reliability and efficiency and possesses the extra desirable property of being network coding compatible. The inner code is formed by linear network coding for combating packet loss, improving the multicast efficiency, etc. A detailed design and analysis of BATS codes are provided in BATSMonograph [BATSMonograph].

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Procedures

2.1. Introduction

A BATS code scheme includes an outer code encoder (also called encoder), an inner code encoder (also called recoder) and a decoder. The BATS code scheme described in this document can be used by a File Delivery Protocol (FDP) with the following procedures.

Encoding at a source node which has the file for transmission:

* The FDP provides the file to be delivered and the related information to the BATS encoder.

* The BATS encoder generates a sequence of batches, each consisting of a set of coded packets and the information pertaining to the batch.

* The FDP forms and transmits the FDP packets using the batches and the corresponding batch information.

Recoding at an intermediate node that does not need the file:

* The FDP extracts the batches and the corresponding batch information from its received FDP packets.

* A BAST recoder generates recoded packets of a batch.

* The FDP forms and transmits FDP packets using the recoded packets and the corresponding batch information.

Decoding at a destination node that needs the file:

* The FDP extracts the batches and the corresponding batch information from its received FDP packets.

* A BATS decoder tries to recover the transmitted file using the received batches.

* The FDP sends the decoded file to the application that needs the file.

2.2. File Partitioning and Padding

Suppose that the FDP has a file of F octets for transmission. The construction of source packets from the file depends on two parameters K and T:

- K: the number of source packets.
- T: the size of a source packet, in octets.

If F is smaller than T*K, the file needs to be padded to have T*K octets, so that file can be partitioned into K source packets, each of which has T octets.

2.3. File Delivery Procedures

2.3.1. Source Node Procedures

A source node has the file for transmission. The FDP will first pad and partition the file into K source packets, each containing T octets. The FDP provides the BATS encoder with the following information:

- Batch size (M): the number of coded packets in a batch.
- Recoding field size (q): the number of elements in the finite field for recoding.
- The degree distribution (DD), optional.
- A sequence of batch IDs (ID[i],i=0,1,...).
- Number of source packets (K).
- Packet size (T): the number of octets in a source packet.
- The source packets (b[i],i=0,1,...,K-1).

Using this information, the (outer code) encoder generates a batch for each batch ID. For each batch ID, the encoder returns the FDP

- a sparse degree (d), and
- M coded packets (X'[i],i=0,1,...,M-1), each containing T' octets.

Here T' = T + M*ceil(log2(q))/8.

The FDP will use the batches to form FDP packets to be transmitted to other network nodes towards the destination nodes. The FDP MUST deliver with each coded packet its

- sparse degree,
- batch ID and certain extra information so that any receiver of the coded packets of the batch can know whether the coded packets are in the same batch or not, and whether two different batches are generated from the same file or not.

The FDP MUST deliver the following information to each recoder:

- batch size M, and
- recoding field size q.

The FDP MUST deliver the following information to each decoder:

- batch size M,
- recoding field size q,
- the file size F, and
- the number of source packet K.

The packet length information is assumed to be known by all the nodes.

2.3.2. Intermediate Node Procedures

An intermediate node does not need the file, but only helps to deliver the file to the destination nodes. At an intermediate node, the FDP only receives the FDP packets from the other network nodes, and should be able to extract coded packets and the corresponding batch information from these packets.

The FDP provides the recoder with the following information:

- the batch size M,
- the recoding field size q,
- a number of received coded packets of the same batch, each containing T' octets, and
- a number M' of recoded packets to be generated.

The recoder uses the information provided by the FDP to generate M' recoded packets, each containing T' octets. The FDP uses the M' recoded packets to form the FDP packets for transmitting.

2.3.3. Destination Node Procedures

A destination node needs the file transmitted by the source node. At the destination node, the FDP receives FDP packets from the other network nodes, and should be able to extract coded packets and the corresponding batch information from these packets.

The FDP provides the decoder with the following information:

- F: number of octets in the file,
- M: batch size,
- q: recoding field size,
- K: number of source packets
- A sequence of batches with their corresponding batch IDs and degrees.

The decoder uses this information to decode the K source packets. If successful, the decoder returns the recovered K source packets to the FDP, which will use the source packets to form the source file.

2.4. Recommandation for the Parameters

The recommendation for the parameters M, K, T, and T' is shown as follows:

- M is 8, 16 or 32.
- q is 2, 4, 8, 16, 32, 64, 128 or 256.
- T' is not larger than the maximum coded packet payload size.
- T = T' - M*ceil(log2(q))/8.
- K = ceil(F/T).

It is RECOMMENDED that K is at least 128. However, the encoder/ decoder SHALL support an arbitrary positive integer value of K.

2.5. Example FDP Packets

A FDP can form a FDP packet by appending a header and a footer to each coded packets.

The header should include F, M, K, q, batch ID, and degree.

3. BATS Code Specification

3.1. Background

The T octets of a source packets are treated as a column vector of T elements in GF[256]. Linear algebra and matrix operations over finite fields are assumed in this section.

Assume that a pseudo-random number generator Rand() is given.

3.2. Outer Code Encoder

Let b[0], b[1], ...,b[K-1] be the K source packets. A batch with batch ID bID is generated in the following steps.

First, a degree DEG=DEG(bID) is sampled using the give degree distribution and Rand() with the default seed. After that, initialize Rand() with bID as the seed.

Second, using Rand() sample DEG packets among all the source packets. Suppose the indices of the packets sampled are i_1, i_2, ..., i_DEG.

Third, sample a DEGxM generator matrix G.

Fourth, form the batch X = (b[i_1], b[i_2], ..., b[i_DEG])*G, where each column is a coded packet of the batch.

Last, append coefficient vectors to the packets of the batches. Let X[i], i=0,1,...,M-1, be the (i+1)th column of X. The coefficient vector of X[i] is the (i+1)th column of the MxM identity matrix with entries in GF(q), which can be represented by M*log2(q)/8 octets. The coded packet X'[i] is formed by appending the coefficient vector before X[i].

3.3. Inner Code Encoder (Recoder) Recommandations

The inner code comprises (random) linear network coding applied on the coded packets belonging to the same batch. At a particular network node, recoded packets are generated by (random) linear combinations of the received coded packets of a batch. The recoded

packets have the same batch ID, sparse degree and coded packet length.

3.4. Decoder Recommandations

The belief propagation decoding algorithm suggested in the BATS code paper [BATS] is recommanded.

4. IANA Considerations

This memo includes no request to IANA.

5. Security Considerations

Subsuming both Random Linear Network Codes (RLNC) and fountain codes, BATS codes naturally inherit both their desirable capability of offering confidentiality protection as well as their vulnerability towards pollution attacks.

5.1. Provision of Confidentiality Protection

Since the transported messages are linearly combined with random coefficients at each recoding node, it is statistically impossible to recover individual messages by capturing the coded messages at any one or small number of nodes. As long as the coding matrices of the transported messages cannot be fully recovered, any attempt of decoding is equivalent to randomly guessing the transported messages. Thus, with the use of BATS codes, information confidentiality throughout the data transport process is assured.

The only thread towards confidentiality exists in the form of eavesdropping onto the initial encoding process, which takes place at the encoding nodes. In these nodes, the transported files are presented in plain text and can be read along their transfer paths. Hence, information isolation between the encoding process and all other user processes running on the node must be assured.

In addition, the authenticity and trustworthiness of the encoding, recoding and decoding program running on all the nodes must be attested by a trusted authority. Such a measure is also necessary in countering the pollution attacks.

5.2. Countermeasures against Pollution Attacks

Like all network codes, BATS codes are vulnerable under pollution attacks. In these attacks, one or more compromised coding node(s) can pollute the coded messages or inject forged messages into the coding network. These attacks can prevent the receivers from

recovering the transported files correctly. Although error detection mechanisms can be put in place to prevent the receivers from getting the wrong messages, detection and discard of the polluted messages still constitute a form of denial-of-service (DoS) attack.

The research community has long been investigating the use of various signature schemes (including homomorphic signatures) to identify the forged messages and stall the attacks (see Zhao07 [Zhao07], Yu08 [Yu08], Agrawal09 [Agrawal09]). Nevertheless, these counter measures are regarded as being computationally to expensive to be employed in broadband communications. A practical approach to protect against pollution attacks consists of the following system-level countermeasures:

1. Attestation and Validation of all encoding, recoding and decoding nodes in the network. Remote attestation and repetitive validation of a node based on valid public key certificates with proper authorization MUST be a pre-requisite of admitting that node into a network and permitting it to remain in that network.

2. Attestation of all encoding, recoding and decoding programs used in the coding nodes. All programs used to perform the BATS encoding, recoding and decoding processes MUST be remotely attested before they are permitted to run on any of the coding nodes. Reloading or alteration of programs MUST NOT be permitted during an encoding session. Programs MUST be attested or validated again when they are executed in new execution environments instantiated even in the same nodes.

3. Original Authentication of all coded messages using network or transport level secure protocols such as IP-sec or TLS/DTLS MUST be used to provide Peer or Message Origin Authentication to every coded message sent through the coding network.

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

6.2. Informative References

[Agrawal09]
S. Agrawal and D. Boneh, "Homomorphic MACs: MAC-based integrity for network coding", International Conference on Applied Cryptography and Network Security , 2009.

[BATS] S. Yang and R.W. Yeung, "Batched Sparse Codes", IEEE Transactions on Information Theory 60(9), 5322-5346, 2014.

[BATSMonograph]
S. Yang and R.W. Yeung, "BATS Codes: Theory and Practice", Morgan & Claypool Publishers , 2017.

[RFC6330] Luby, M., Shokrollahi, A., Watson, M., Stockhammer, T., and L. Minder, "RaptorQ Forward Error Correction Scheme for Object Delivery", RFC 6330, DOI 10.17487/RFC6330, August 2011, <https://www.rfc-editor.org/info/rfc6330>.

[Yu08] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An Efficient Signature-Based Scheme for Securing Network Coding Against Pollution Attacks", NFOCOM , 2008.

[Zhao07] F. Zhao, T. Kalker, M. Medard, and K.J. Han, "Signatures for content distribution with network coding", ISIT , 2007.

Appendix A. Additional Stuff

This becomes an Appendix.

Authors' Addresses

Shenghao Yang
CUHK(SZ)
Shenzhen, Guangdong
China

Phone: +86 755 8427 3827
Email: shyang@cuhk.edu.cn

Xuan Huang
CUHK(SZ)
Shenzhen, Guangdong
China

Phone: +86 755 8427 3814
Email: 115010159@link.cuhk.edu.cn

Raymond W. Yeung
CUHK
Hong Kong, Hong Kong SAR
China

Phone: +852 3943 8375
Email: whyeung@ie.cuhk.edu.hk

John Zao
NCTU
Hsinchu, Taiwan
China

Email: jkzao@ieee.org