An Efficient Algorithm for ML Decoding of Raptor Codes over the Binary Erasure Channel

Saejoon Kim, *Member, IEEE*, Seunghyuk Lee, and Sae-Young Chung, *Senior Member, IEEE*

***Abstract*—In this letter, we propose an efficient algorithm for maximum-likelihood decoding of Raptor codes used over the binary erasure channel. The algorithm is inspired by the decoding scheme suggested in the 3GPP Multimedia Broadcast/Multicast Services standard and is an improved version of it.**

***Index Terms*—Raptor codes, maximum-likelihood decoding, erasure channels.**

I. INTRODUCTION

RAPTOR codes [2] are rateless error-correcting codes that are universally capacity-achieving over the binary erasure channel (BEC). Given a set of message symbols, Raptor codes generate encoding symbols on-the-fly and are able to recover the set of message symbols perfectly given any set of encoding symbols whose cardinality is only slightly greater than that of the message symbols. Furthermore, Raptor codes can be encoded in linear-time and have decoding algorithm that runs in linear-time which makes them very appealing for various applications requiring loss protection. For these reasons, Raptor codes have recently been adopted by the 3GPP standard as the forward error correction scheme in Multimedia Broadcast/Multicast Services (MBMS) [3].

While Raptor codes used over the BEC can either be decoded by the linear-time but suboptimal message passing algorithm [1] or the maximum-likelihood (ML) algorithm, it is known that Raptor codes decoded by the ML algorithm significantly outperforms those decoded by the message passing algorithm for finite message lengths. Presumably for this reason, the 3GPP standard [3] outlined a time-efficient ML decoding scheme that can be implemented in real-time even with limited computing capability for message lengths supported by the standard. In this letter, we propose an improved version of the time-efficient ML decoding scheme sketched in the standard. Our proposed algorithm runs considerably faster than the scheme in the standard and in particular, is able to run more than 8600 times faster than off-the-shelf Gaussian elimination algorithm for ML decoding of Raptor codes of the 3GPP standard.

II. SYSTEMS AND METHODS

Raptor codes are a serially concatenated codes with a pre-code as the outer code and the LT code [1] as the inner code. We shall consider here the systematic Raptor codes of the 3GPP standard in which the pre-code itself is also a serially concatenated code with an LDPC code as the outer code and a code with dense parity check matrix as the inner code. In this Raptor code, a k-symbol message $\mathbf{m}$ is first encoded by the rate-$\frac{k}{k+s+h}$ pre-code to a vector of $k+s+h$ intermediate symbols $\hat{\mathbf{m}}$ which is then encoded by the LT code to a stream of encoding symbols. Here, s and h are the amount of redundancy added by the outer and the inner codes of the pre-code, respectively. Suppose $\mathbf{c}$ is the vector representing n received encoding symbols and $\hat{\mathbf{c}}$ is the concatenation of 0^{s+h} and $\mathbf{c}$ where 0^{s+h} is the all-zero symbol string of length $s+h$. Then $\hat{\mathbf{m}}$ and $\hat{\mathbf{c}}$ are related by $A\hat{\mathbf{m}}^t = \hat{\mathbf{c}}^t$ where A is an $n+s+h$ by $k+s+h$ matrix and the superscript t represents the transpose operator. In this matrix A, the top $s+h$ rows represent the constraints of the pre-code on $\hat{\mathbf{m}}$, and the bottom n rows, each corresponding to a received encoding symbol of $\mathbf{c}$, represent the generator matrix of the LT code. Raptor codes in the 3GPP standard were intricately designed so that once $\hat{\mathbf{m}}$ is found, the original message $\mathbf{m}$ can be computed very efficiently. Moreover, since decoding succeeds if and only if $\hat{\mathbf{m}}$ can be found, decoding Raptor codes essentially reduces to that of calculating the set of intermediate symbols $\hat{\mathbf{m}}$.

A. 3GPP MBMS Decoding

The decoding scheme suggested in the 3GPP MBMS standard is a time-efficient version of the computationally expensive "standard" Gaussian elimination decoding and consists of four phases after which the original matrix A will be converted into a diagonal matrix. In the first phase, which is the key phase contributing to fast decoding, the goal is to convert the matrix A into a matrix consisting of left upper and lower and right submatrices. In this matrix, the left upper submatrix is a diagonal matrix and the left lower submatrix is an all-zero matrix. The right submatrix U consisting of the last u columns of A is a don't-care matrix for now. A key ingredient in the first phase is the submatrix V which is formed by the intersection of all but the first i columns and the last u columns and all but the first i rows of A for nonnegative integers i and u. At the beginning of the first phase, i and u are equal to 0, and hence $V = A$. The matrix V will change during the course of the first phase at the end of which will disappear if the phase is successful.

Before describing the first phase of decoding, consider when the minimum number of 1's in any row in V is two and the graph G which is constructed as follows. Let the columns of A that intersect V be the set of nodes of G and the rows that have exactly two 1's in V be the set of edges of G that connect the associated nodes. A *component* of G is a set of nodes and edges in G in which there is a path between all

Manuscript received April 16, 2008. The associate editor coordinating the review of this letter and approving it for publication was G. Taricco.

S. Kim and S. Lee are with the Department of Computer Science and Engineering, Sogang University, Seoul, Korea (e-mail: saejoon@sogang.ac.kr).

S.-Y. Chung is with the Department of EECS, KAIST, Daejeon, Korea (e-mail: sychung@ee.kaist.ac.kr).

Digital Object Identifier 10.1109/LCOMM.2008.080599.

1089-7798/08$25.00 © 2008 IEEE

pairs of nodes, and the size of the component is the number of nodes in the component. 3GPP decoding will invoke the notion of maximum size component in its first phase in an effort to minimize the decoding time in its remaining phases.

The following describes the first phase of 3GPP decoding where r represents the minimum positive value where at least one row has exactly r 1's in V, and by abuse of terminologies, a row or column of V represents a row or column of A that intersects V, respectively.

First Phase of 3GPP Decoding

Step 1) Choose a row in A with r 1's in V.
- a) When $r \neq 2$, choose any such row.
- b) When $r = 2$, choose any such row contained in the maximum size component of the graph G associated with V.

Step 2) The first row of V is exchanged with the chosen row and the columns of V are reordered such that the column of V of one of the r 1's in the chosen row is exchanged with the first column of V and the columns of remaining $r-1$ 1's are exchanged with the last columns of V.

Step 3) All rows of V that have a 1 in the first column of V are exclusive-ORed with the first row of V so that all rows below it have a 0 in the first column of V.

Step 4) Value of i is incremented by 1 and that of u is incremented by $r-1$.

Step 5) Go to Step 1).

This phase succeeds when $i + u = k + s + h$ and fails otherwise. In other words, this phase fails when all entries of V become 0 before it disappears. Note that whenever row operations or row or column exchanges are made in the matrix A, the corresponding changes must also be made in the vectors $\hat{\mathbf{m}}$ and $\hat{\mathbf{c}}$ to maintain equality in $A\hat{\mathbf{m}}^t = \hat{\mathbf{c}}^t$. An important goal of the first phase is to minimize the number of columns of U which increases only when $r \geq 2$. Observe that when a row in a component is chosen, then all rows in the component will be chosen for processing sequentially in the next steps via chain reaction. For example, if the size of the component is γ, then the value of i is incremented by $\gamma - 1$ and that of u is incremented by 1 after all rows in the component are processed. Hence, choosing a row contained in the maximum size component whenever $r = 2$ can be viewed as a greedy method to minimize the number of columns of U.

Let U_1 and U_2 denote the first i and the remaining rows of the submatrix U. In the second phase, a standard Gaussian elimination is applied to U_2 to convert it into a matrix in which the first u rows is the identity matrix. This part of the decoding explains why a row in the maximum size component is chosen for processing when $r = 2$. If the Gaussian elimination is successful, then the bottom $n-k$ rows of A are discarded and the obtained matrix at the end of phase two is a square matrix of size $k+s+h$ with 1's along the diagonal. If this is not possible which is the case when the rank of U_2 is less than u, then second phase fails. Third and fourth phases are relatively much simpler than the first two phases and therefore require significantly much less processing time. In these phases, the submatrix U_1 is converted into an all-zero matrix through exclusive-OR's with the bottom u rows after which A becomes the identity matrix of size $k+s+h$. We note that the success of the described decoding scheme depends only on that of the first and second phases.

B. Proposed Decoding Scheme

A computational bottleneck in 3GPP decoding lies in the computation of the maximum size component of G which requires $O(n+\nu)$ basic operations where ν is the number of edges in G. Our proposed decoding scheme is a heuristic effort to remedy this problem and is identical to the 3GPP decoding except when $r \geq 2$ in Step 1 of the first phase of the decoding. Our proposed decoding scheme defines the *score* of the i^{th} row as the number of 1's that is removed from V after the i^{th} row is processed. Interpreting the matrix $A \triangleq (a_{ij})$ as a bipartite graph with $k+s+h$ left nodes and $n+s+h$ right nodes so that j^{th} left node is connected to i^{th} right node if and only if $a_{ij} = 1$, the score of an i^{th} row is just the sum of the degrees of the i^{th} right node's neighbors. Note that in this bipartite graph representing the matrix A, the degree of each node which is a constant and the indices of neighbors of each node are stored throughout the course of decoding anyway in 3GPP decoding. Therefore, the computation of the score for each row requires only a fixed amount of basic operations which implies that only $O(n)$ basic operations are sufficient to find the row with the highest score. The following describes Step 1 of the first phase of our proposed decoding scheme.

Selection of a Row in A with r 1's in V

- a) When $r = 1$, choose any such row.
- b) When $r \geq 2$, choose any such row with the highest *score*.

Simple calculation shows that Steps 2 $\sim$ 4 in the first phase of the decoding can be completed with a fixed amount of basic operations, and since more than 90% of the rows in A are already processed typically before $r = 2$ in decoding Raptor codes of the 3GPP standard, the first phase of our proposed decoding scheme runs in near $O(n)$-time. Moreover, since the number of columns created in U is also very small for the Raptor codes, the processing time of the standard Gaussian elimination in phase two of the decoding is also near $O(n)$ facilitating the processing of all four phases in our proposed decoding scheme to run in near $O(n)$-time.

The weakness of using the score as the criteria for row selection is that the number of columns of U created can be slightly larger than in the 3GPP case, however, considerably smaller than if a row with two 1's in V were to be selected at random. As will be shown in the next section, the number of columns of U created through this approach is only slightly larger than through the 3GPP approach, and in particular, the advantage of using the score as the criteria for row selection when $r = 2$ outweighs its weakness.

If the number of received encoding symbols is much larger than that of the message symbols, say, $n \geq 1.3k$, then there will almost always be rows present with exactly one 1 in V in the Raptor code of the 3GPP standard. In other words, for a sufficiently large number of received encoding symbols both the 3GPP and our proposed decoding scheme reduce to the message passing algorithm, *i.e.*, $u = 0$ at the end of the first phase. Therefore, to effectively test the efficiency of the Gaussian elimination decoding of the 3GPP standard and our proposed scheme, a small reception overhead need be chosen.

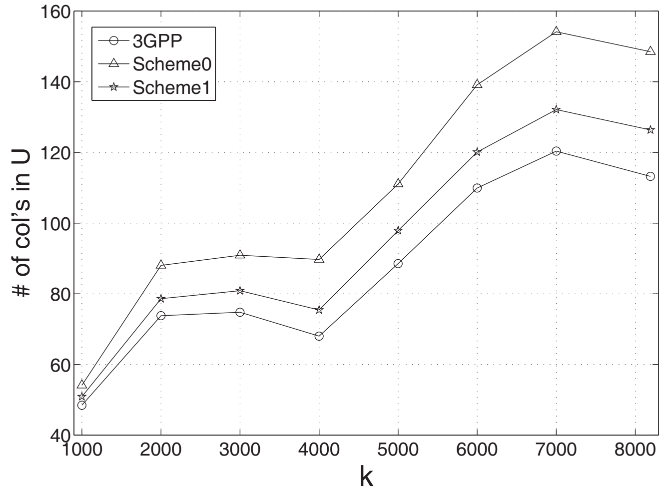

Fig. 1. Number of Columns in Submatrix U.

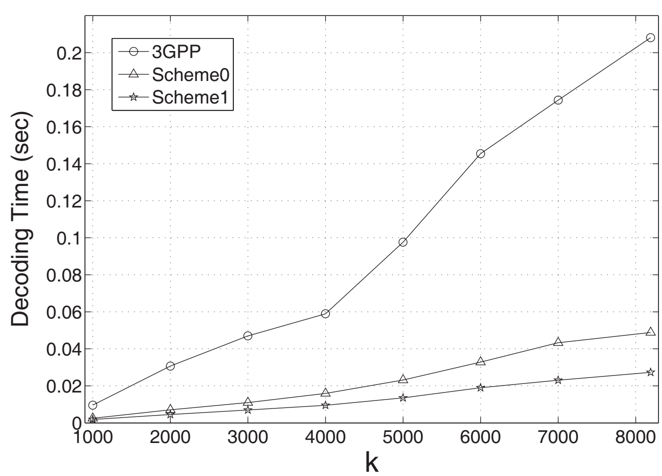

Fig. 2. Decoding times in seconds.

III. Results

In this section, we present the decoding performances obtained using the 3GPP decoding scheme (3GPP) and our proposed decoding scheme (Scheme 1) tested on the Raptor codes of the 3GPP standard. To quantitatively assess a comparison of the number of columns in U created and the decoding time elapsed between the two decoding schemes which are identical except Step 1 of the first phase of the decoding, we also tested a decoding scheme (Scheme 0) identical to 3GPP decoding except Step 1 of the first phase of decoding that is replaced by choosing a row with r 1's in V at *random* for all values of r. All simulations in this letter were tested on Intel(R) Xeon(R) CPU @1.60GHz and the results are based on 10000 runs with reception overhead equal to 1% for message lengths $k = 1000, 2000, \cdots, 7000$ and 8192.

Figure 1 shows the average number of columns in U created with respect to the number of message symbols k after a successful first phase by the three decoding schemes. As expected, 3GPP clearly created the least number of columns in U and our proposed scheme of Scheme 1 created noticeably smaller number of columns in U than the random row selection scheme of Scheme 0. Note that while the curves in Fig. 1 would expect to increase monotonically, the curves decrease when $k = 4000$ and 8192. We discovered that this behavior is an artifact of the Raptor codes of the 3GPP standard which arises because there are more 1's in the top $s+h$ rows in the matrix A when k takes values around 4000 and 8000 compared to other values of k. In contrast to intuition, rows with many 1's in the pre-code part of the matrix A are more likely to become low-weight rows during the processing of the first phase of decoding and eventually not contribute to increasing the value of u.

Figure 2 shows the average decoding time elapsed in seconds with respect to the number of message symbols k after successful decoding completion by the three schemes. The curves in the figure were generated using an efficient implementation of the three decoding schemes and the only part that is different in the implementations is the Step 1 of the first phase of the decoding. The figure indicates that Scheme 1 has the best decoding time performance while Scheme 0 which created the most number of columns in U performs significantly better than 3GPP which created the least number of columns in U. The relative decoding time performance between Scheme 1 and Scheme 0 can be explained by the fact that both have the same processing time requirements of $O(n)$ in Step 1 of the first phase of decoding while Scheme 1 has a smaller number of columns in U to process in the remaining phases of decoding. For 3GPP, the figure implies that the search for the maximum size component in an effort to minimize the number of columns in U was too costly for the savings gained in the number of columns in U compared to both Scheme 1 and Scheme 0. For example, Scheme 1 runs approximately ten times faster than 3GPP for all messages lengths supported by the 3GPP standard. We note that if one were to apply standard Gaussian elimination algorithm to decode Raptor codes of the 3GPP standard, then about 237 seconds is required to decode $k = 8192$ codes which is more than 8600 times what is required by Scheme 1.

IV. Discussion

In this letter, a time-efficient algorithm for ML decoding of Raptor codes used over the BEC was presented. The presented algorithm is a modified version of the decoding scheme outlined in the 3GPP standard and was demonstrated to be approximately ten times faster than the algorithm of the 3GPP standard. Furthermore, it was shown that the proposed decoding algorithm runs in near linear-time.

Acknowledgement

The authors wish to thank the anonymous reviewers for detailed comments which greatly helped to improve the presentation of the paper.

References

[1] M. Luby, "LT codes," in *Proc. 43rd Annual IEEE Symp. Foundations of Computer Science*, Vancouver, Canada, 2002.

[2] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[3] 3GPP TS 25.346 v.7.0.0, 2006, Technical Specification Group Radio Access Network; Introduction of the Multimedia Broadcast/Multicast Services (MBMS) in the Radio Access Network, Mar. 2006.