The Throughput of LDPC-Based Incremental-Redundancy Schemes with Finite Blocklength

Stefania Sesia
Eurecom Institut - Mobile Dep.
2229 Route des Crêtes
06904 Sophia Antipolis Cedex, France
e-mail: Stefania.Sesia@eurecom.fr

Giuseppe Caire
Eurecom Institut - Mobile Dep.
2229 Route des Crêtes
06904 Sophia Antipolis, France
e-mail: Giuseppe.Caire@eurecom.fr

Guillaume Vivier
Centre de Recherche Motorola
Espace Technologique de S. Aubin
91193 Gif-sur-Yvette, France
e-mail: Guillaume.Vivier@crm.mot.com

I. Introduction

We consider a frequency-flat block-fading Gaussian channel where transmission is slotted and the channel gain over each slot is random but constant for the whole slot. For simplicity, we assume that the channel gains over different slots are statistically independent and Rayleigh distributed. In this work, we consider the very powerful class of *irregular* Low-Density Parity-Check (LDPC) codes, [2], and show that they are very good candidates for efficient Hybrid ARQ (H-ARQ) schemes. The H-ARQ scheme under analysis is referred to as Incremental Redundancy (IR) protocol [1]. In the paper we will show the throughput of the IR protocol for certain random coding ensembles and deterministic LDPC code constructions.

II. Results

Finite blocklength LDPCs show a performance loss in terms of throughput with respect to the infinite length counterparts, which indeed achieve near-optimal throughput. Two approaches to improve the performance of IR with finite length practical LDPC codes have been envisaged [3]. The first approach acts directly on the code design and leaves the IR protocol unchanged. It consists of selecting the code parity-check matrix in some appropriate ensemble with good FER properties (labeled as "Modified Ensemble" in the Figure). The second approach acts on the IR protocol and leaves the code design unchanged. It stems from the fact that for standard irregular LDPC codes, most frame errors involve a very small number of bit errors. Therefore, by dividing the information packet into smaller subpackets, only a few of them will contain errors after decoding. Hence, an Outer Selective-Repeat (OSR) protocol acting on these smaller subpacket units can recover subpacket errors without having to retransmit the whole codeword. Thus, only the erroneous packets are retransmitted together with new information packets. We shall consider "successful" decoding (i.e., the IR protocol stops the transmission of the current codeword) if the fraction of subpackets in error is less than a threshold value δ. Otherwise, a NACK is sent and the next block of the current codeword is transmitted on the next slot. The optimistic assumption underlying this approach is that subpackets in error can be perfectly detected. The system throughput can be optimized with respect to the threshold δ. We have analytically computed the throughput of the IR-OSR protocol for a fixed value of δ. However, since analytical optimization of δ is difficult, if not impossible, we exhaustively searched for the best threshold value.

Figure 1 shows the throughput as a function of $\delta \in [0,1]$ and $\gamma = 10$dB where the subpacket length of the OSR protocol is fixed to 6 bytes, the length of the code equals 10000 bits and the rate equals 0.32bit/symbol. We notice that the performance of the OSR is quite insensitive to the value of δ (unless δ is either very close to 0 or very close to 1). We also plotted the throughput achieved by the same ensemble with infinite length, with finite length without any countermeasure and with finite length by averaging over the modified ensemble. These results are shown as horizontal lines as they do not depend on δ. Interestingly, although these approaches are quite different, they yield almost the same performance improvement and recover a considerable fraction (up to 80% at SNR = 10 dB) of the loss due to finite with respect to infinite length. Moreover, combining the two methods does not provide additional gain. This fact can be explained by noticing that for a typical code in the modified ensemble a frame-error corresponds to a large number of bit errors (i.e., a large number of subpackets to retransmit). Hence, using the OSR protocol does not improve the throughput [3].

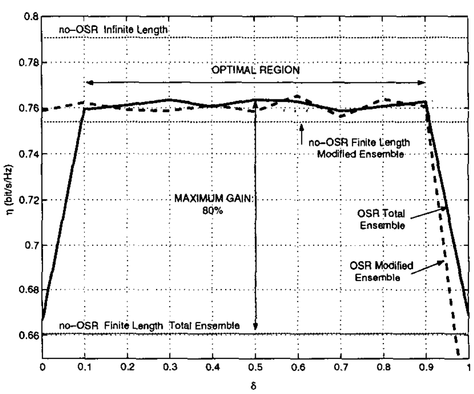

Figure 1: Throughput vs δ for $\gamma = 10$dB, $R = 0.32$bit/symbol, blocklength $n = 10000$ with OSR and without OSR (labeled "no-OSR").

References

[1] Caire G. and D. Tuninetti, "The Throughput of Hybrid-ARQ Protocols for the Gaussian Collision Channel", *IEEE Trans. on Information Theory*, Vol. 47, pp. 1971-1988, July 2001.

[2] T.J. Richardson and R.L. Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding", *IEEE Trans on Information Theory*, Vol. 47, pp. 599-618, February 2001.

[3] S. Sesia, G. Caire, and G. Vivier "Incremental Redundancy Hybrid ARQ Schemes Based on Low-Density Parity-Check Codes", *Submitted to IEEE Trans. on Communications*, February 2003.

0-7803-7728-1/03/$17.00 ©2003 IEEE.