On Reversible Variable Length Codes with Turbo Codes, and Iterative Source-Channel Decoding

Ksenija Laković and John Villasenor
University of California, Los Angeles
Los Angeles, CA 90095-1594
e-mail: {ksenija,villa}@icsl.ucla.edu

I. Introduction

In recent years, increased attention has been given to the design of variable length codes (VLCs) for efficient and reliable communications. With regard to traditional source code design criteria, e.g. average code length and efficiency, Huffman codes are known to be optimal instantaneous source codes. However, alternative classes of VLCs, such as reversible variable length codes, exhibit better robustness to transmission errors. As the name suggests, reversible variable length codes (RVLCs) allow instantaneous decoding in both directions, which can be used for error recovery by combined forward and backward processing [1]. The bi-directional processing with RVLCs yields improved image quality in video compression standards (H.263 and MPEG-4).

RVLCs are also advantageous over Huffman codes when source decoding is performed by the Viterbi algorithm, i.e. by use of the soft-input trellis-based decoder [2]. The improved performance results from better distance properties of RVLCs. Furthermore, in combination with convolutional channel codes, RVLCs outperform Huffman codes when non-iterative decoding is used, and the improvement can be further increased with iterative source-channel decoding [2].

Motivated by the good performance of RVLCs and by their growing popularity in video coding standards, we consider RVLCs in combination with turbo codes. We discuss RVLC-turbo coding relative to Huffman-turbo coding, and propose a new source-channel decoder that exploits the RVLC structure during the iterative decoding process. We show that the introduced method yields significantly better performance than other VLC-turbo decoding systems.

II. Source-Channel Coding with VLCs and Turbo Codes

Consider a two-stage coding system that consists of a variable length source code followed by a turbo-code. This system achieves optimal performance if two conditions are met. First, the VLC should provide the optimal binary representation of source symbols, which means that its average code length (L_{avg}) should be equal to the source entropy ($H(U)$). This condition can be expressed as $E_s = H(U) \cdot E_b$, where E_s represents the energy per source symbol (at the input of the VLC), and E_b the energy per information bit (at the output of the VLC). Second, assuming the transmission over a Gaussian channel, the turbo code of rate R information bits per channel use should yield a zero error-rate at a signal to noise ratio higher than $E_b/N_o = (2^{2R} - 1)/2R$. Therefore, the optimal source-channel coding scheme should achieve a zero error-rate at an SNR higher than $E_s/N_o = H(U) \cdot (2^{2R} - 1)/2R$. However, there are limitations associated with VLCs and turbo codes that preclude the optimal performance. With VLCs, the average code length depends on the source distribution, and is seldom equal to the entropy. Turbo codes achieve good performance at a threshold (about 0.5-1dB) away from the optimal SNR, and exhibit a non-zero error-rate ("the error floor") even at a relatively high SNR levels.

Assume first that source and channel decoding are performed separately, i.e. that after i turbo decoding iterations the soft-output of the turbo decoder is fed to the VLC decoder. In this case, the residual source code redundancy remains unused during the iterative decoding process, and the iterative convergence threshold depends only on the channel code and E_b/N_o. For a non-optimal VLC, it holds that $E_s = E_b \cdot L_{avg}$, which implies that a longer L_{avg} causes the decoding convergence at a higher E_s/N_o. Therefore, in terms of decoding convergence, it is preferable to use a VLC of the shortest possible code length. On the other hand, in terms of the error floor, it is advantageous to use a VLC with good error-correcting properties. While a few turbo decoding errors at the input of an error-sensitive VLC typically cause a burst of errors at its output, a more robust VLC is usually capable of correcting some errors and eliminating the bursty effect. In conclusion, in the separate source-channel decoding scenario, RVLCs are advantageous over Huffman codes with regard to the error floor, but disadvantageous with regard to decoding convergence.

However, it is possible to construct a turbo-VLC decoding scheme that exploits VLC structure during the iterative decoding process. We have developed a number of such schemes, one early example of which was described in [3]. Iterative source-channel decoding with turbo codes and RVLCs utilizes error-correcting properties of RVLCs, and therefore yields good performance at all SNR levels. Simulations results that verify this can be found in [3].

III. Conclusions

We have studied the emerging class of robust source codes, reversible variable length codes, in combination with turbo codes. Relative to Huffman codes, RVLCs are advantageous at a high SNR (in the "error floor region") due to better error-correcting properties. However, in the separate turbo-VLC decoding scenario, Huffman codes are advantageous over RVLCs at a relatively low SNR (in the "waterfall region"). In this scenario, the higher redundancy of RVLCs is unused during the iterative decoding process. We have proposed a joint iterative turbo-RVLC decoder, which utilizes RVLC structure, and is advantageous over other decoding systems at all SNR levels.

References

[1] Y. Takishima, M. Wada, H.Murakami, "Reversible variable length codes," *IEEE Trans. Commun.*, vol.43, (no.2-4), 1995.

[2] R. Bauer, J.Hagenauer, "On variable length codes for iterative source/channel-decoding," *IEEE DCC*, March 2001.

[3] K. Laković, J. Villasenor, "Combining Variable Length Codes and Turbo Codes," *IEEE VTC*, May 2002.