Joint source-channel coding with adaptation

Minh Quang Nguyen, Hang Nguyen, Eric Renault, Phan Thuan Do

To cite this version:

Joint source-channel coding with adaptation

Minh-Quang Nguyen[1,2], Hang Nguyen[1], Eric Renault[1], Phan-Thuan Do[2]
[1] SAMOVAR, Télécom SudParis, CNRS, Université Paris-Saclay, 9 rue Charles Fourier - 91011 Evry Cedex
{minh_quang.nguyen, hang.nguyen, eric.renault}@telecom-sudparis.eu
[2]School of Information and Communication Technology, Hanoi University of Science and Technology
Email: thuandt@soict.hust.edu.vn

***Abstract*—According to Shannon's source-channel separation theorem, the source coding and the channel coding operations should be performed independently. However, such a method does not exploit the nature of unequiprobable source output in a finite block regime so that it can not achieve good qualities in several scenarios in practical systems. In this study, we propose a JSCC strategy which takes into account the unequiprobable source output to attain better data quality than traditional methods. An adaptation is performed in order to map source code outputs to appropriate inputs of channel code. We also propose a greedy algorithm to construct parameters for the adaptation in polynomial time. Our simulation is performed on AMR compression to estimate the effect of our approach as compared to the traditional strategy. Using this approach we find significantly improved speech quality after transmission in both cases: transfer of only one AMR codec mode and transfer of all AMR codec modes.**

***Index Terms*—joint source channel coding, Shannon thoery, information theory, speech, unequal error protection**

I. Introduction

Joint source-channel coding (JSCC) is considered as one of the enabling technologies for reliable communications in which a source code and a channel code are performed sequentially on data before transmission over noisy channels. During source coding, we choose the most effective representation of data to remove all the redundancy and form the most compressed version possible. Conversely, during channel coding, we add redundant information to the compressed data prior to data transmission, and a receiver can then recover original data which contains no apparent errors. In practical systems, a perennial question is how to design a source code and a channel code that can transfer data effectively.

The relationship between the source code and the channel code was first mentioned in Shannon theory [**?**] which provided a remarkable insight for optimal transmission systems. It includes two parts. The direct part stated that a source can be transmitted over channels in a reliable way if the source coding rate is strictly below the channel capacity. The converse part stated that if the source coding rate is smaller than the channel capacity, reliable transmission is impossible. Shannon's theory shows the limit of what is achievable in the case of infinite time to transmit information. According to the proof of Shannon theorems about source code and channel code, traditional communication systems perform source coding and channel coding separately and independently to transfer data.

Shannon theory shows the limits of what is achievable in the case where time is infinite, i.e. the blocklength of channel code output goes to infinite. In real world systems, especially in modern real time applications, because of their finite blocklength, one cannot wait forever. Kostina et al. [**?**] show that data in practical systems can be transferred more efficiently than traditional methods by choosing an appropriate source and channel code: a source code is chosen with knowledge of the channel, and a channel code is chosen with knowledge of the source's distribution and distortion measurement. In their work, a bounds to show that k, the maximum number of source symbols transmissible using a given channel blocklength n, must satisfy

$$nC - kR(d) = nV + \sqrt{kV(d)}Q^{-1}(\epsilon) + O(logn)$$

under the fidelity constraint of exceeding a given distortion level d with probability ϵ. Where Q is the standard Gaussian complementary cumulative distribution function, $C, V, R(d), V(d)$ are the channel capacity and dispersion, rate-distortion and the rate-dispersion function respectively. This is a remarkable insight, but to achieve the fundamental limit, a lot of work must be done to design codes reaching that limit.

There are many approaches to improve the traditional method in different scenarios. One of the main trends of JSCC is using Unequal Error Protection (UEP). It is based on an assumption: not all information bits are equally important, due to the different sensitivity of the source decoder to errors. UEP consists of allocating coding redundancy depending on the importance of the information bits. In published literature, mainly two different channel packet formats have been considered: variable-length channel packets with fixed-length information block (fixed-k approach) [**?**], fixed-length channel packets with variable length information block (fixed-n approach) [**?**] and variable k, n for different packets (variable-(n, k) approach) [**?**], [**?**], [**?**]. However, little attention has been paid to exploit the unequiprobable distribution and the unequal importances of source code output values.

In this paper, we extend the concept of UEP: instead of differing redundancy allocated to information bits, we map important source output values to appropriate channel input values which are less distorted by the noisy channel than others. Hence, by adding the same redundancy to information bits, important parts of source output are protected more efficiently than others. Simulation results are compared in terms of quality of speech data with the traditional method.

II. Joint source channel coding in finite blocklength

A. Asymptotic optimality

Given a channel blocklength n, a source blocklength k, a rate distortion measurement $R(d)$ and a channel capacity C. The output of the most advantageous source encoder is approximately equiprobable over a set of roughly $\exp(kR(d))$ distinct messages (for large k). The use of this source encoder enables to represent most of the source outcomes within distortion d. On the other hand, the channel coding theorem shows that there exists a channel code that can distinguish $M = exp(kR(d)) < exp(nC)$ messages with high probability if a maximum likelihood decoder is equipped in the system. Therefore, it is shown in the JSCC theorem that one can achieve high-quality transmission (in which there is small probability of distortion exceeding d) by a simple combination of both source and channel codes. However, in practical systems where n is finite, the output of the source encoder is not always equiprobable. Hence schemes that perform source decoding and channel decoding separately without exploiting unequal message probabilities may not achieve near-optimal non-asymptotic performance. In the nonasymptotic regime, there is a need to design a intelligent code that take into account the residual encoded source redundancy at the channel decoder.

B. Definitions

For practical systems where blocklengths are limited, with a specific source code and channel code equipped, we propose a method to take into account the nature of the channel and the unequiprobable source outputs to improve traditional JSCCs. In source coding, there are several values in the compressed data that are more important than others: values which appear in headers regularly or values with high probability distribution. On the other hand, there are several values of the channel code output which are less distorted than others after transfer over a noisy channel. We introduce here a new concept called adaptation, which is considered as a bijection between source code outputs and channel code inputs. The adaptation is performed at the transmitter and the receiver in order to map important source output values to channel code inputs which are less distorted by the channel.

Unlike traditional UEP strategies where protection is brought about based on the unequal importance of data positions in compressed data, our proposed adaptation capitalizes on the unequal importance of source output values. Therefore, the indispensable values are masked by those which are less distorted while being transferred. Moreover, instead of different redundancy amount allocated to compressed data, we use the same channel code rate for every source output values. As a consequence, we only need to use one channel code to protect data.

Assume that we have a source that produces sequences $X \in \mathcal{X}$. The output messages of source code are indexed by the set $\{1,..,M\}$. Before entering the channel decoder, each source

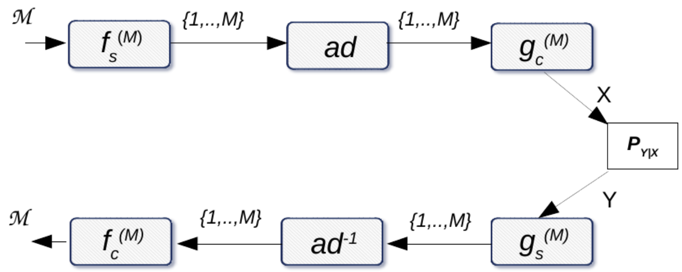

Fig. 1. JSCC with adaptor

output $w \in \{1,..,M\}$ is put into the adaptation which gives $\hat{w} = ad(w)$. At the receiver, a converse process is performed: before source decoding, output w of the channel decoder is put into the adaptation ad^{-1}. The mathematical model of our proposed JSCC is described in Definition 1.

Definition 1 Joint source-channel code with adaptation is a code such that the encoder and decoder mappings satisfy:

$$f = f_c^{(M)} \times ad \times f_s^{(M)}$$

$$g = g_c^{(M)} \times ad^{-1} \times g_s^{(M)}$$

where

$$f_s^{(M)} : \mathcal{M} \to \{1,..,M\}$$

$$ad : \{1,..,M\} \to \{1,..,M\}$$

$$f_c^{(M)} : \{1,..,M\} \to \mathcal{X}$$

$$g_c^{(M)} : \mathcal{Y} \to \{1,..,M\}$$

$$ad^{-1} : \{1,..,M\} \to \{1,..,M\}$$

$$g_s^{(M)} : \{1,..,M\} \to \hat{\mathcal{M}}$$

For a channel characterized by $P_{Y|X}$ and a system equipped with a specific channel code, there is an adaptation ad which maps source code outputs to channel code inputs to attain the least distortion. Work of our proposed JSCC with adaptor is described in Figure 1.

In lossy compression, if compressed data is decompressed without transferring over channel (i.e. it does not suffer any error) the rate distortion is greater than 0. If it is transferred over a noisy channel before being decompressed, errors caused by the channel make the rate distortion greater than that of decompressed data without transmission error. We define the quality degradation of data after transferring the noisy channels as follow:

Definition 2 Let $w_1 \in \{1,...,M\}$ be the outputs of the source code from an input m; $w_2 \in \{1,...,M\}$ be a received value after channel coding and transfering w_2 over channel. With a specific source decoder, the rate distortion augmentation of w_1 to w_2 is given by:

$$s(w_1, w_2) = d(g_s(w_1), m) - d(g_s(w_2), m)$$

where $d(\hat{m}, m)$ is the rate distortion between decompressed message $\hat{m}$ and original message m. Rate distortion augmentation can be interpreted as the difference of the rate distortion when source code output w_1 of m is transformed to w_2 by the noisy channel.

Definition 3 With a JSCC in a finite blocklength mode, we define the probability of receiving a message $\hat{w}$ after transfering over channel and channel decoding when the input of the channel encoder is w:

$$p(\hat{w}|w) = P_{Y|X}(y|x = f_c(w))$$

where $\hat{w} = g_c^{(M)}(y)$.

There are many ways to map the set M to itself to form an adaptation. Adaptations have different effects. To estimate the effect of an Adaptations, we calculate the expectation of the rate distortion augmentation by the formula below:

$$\mathsf{E} = \sum_{w \in \{1,\dots,M\}} \sum_{\hat{w} \in \{1,\dots,M\}} p(w).p(\hat{w}|\bar{w}).s(w, ad^{-1}(\hat{w}))$$

where w is the output of the source coder and $\hat{w}$ is the output of the channel decoder. Based on this formula, we proposed 2 different algorithms to find appropriate adaptations in the next section.

III. Algorithms for the adaptation

A. Adaptation for one-value-protection

The purpose of an adaptation in this scenario is to minimize the rate distortion augmentation which is caused by errors appearing on a particular value $\boldsymbol{w} \in \{1, \dots, M\}$. If $\boldsymbol{w}$ is assigned to $\bar{\boldsymbol{w}}$, i.e. $ad(\boldsymbol{w}) = \bar{\boldsymbol{w}}$ and $ad^{-1}(\bar{\boldsymbol{w}}) = \boldsymbol{w}$, the expectation of rate distortion augmentation caused by errors on $\boldsymbol{w}$ becomes:

$$E_{er(\boldsymbol{w})} = \sum_{\hat{w} \in \{1,\dots,M\}} p(\boldsymbol{w}).p(\bar{\boldsymbol{w}}|\hat{w}).s(\boldsymbol{w}, ad^{-1}(\hat{w}))$$

To find the optimal adaptation in this situation, we solve the optimization problem:

$$\min_{ad(\boldsymbol{w}) = \hat{\boldsymbol{w}}} E_{er(\boldsymbol{w})}$$

The minimization is over all possible assignments for $ad(w), w \in \{1, .., M\}, w \neq \boldsymbol{w}$.

Lemma 1. *Fix $ad(w) = \hat{w}$, the optimal adaptation to minimize quality degradation caused by errors on w can be determined by the following formula:*

$$min \sum_{w_2} p(w_1, \hat{w_2}) s(\hat{w_1}, w_1) = \sum_{i=0}^{N} p(w_1, \hat{w_2^i}) s(\hat{w_1^i}, w_1)$$

in which, for $0 \leq i < j \leq N$

$$p(w_1, \hat{w_2}^i) < p(w_1, \hat{w_2}^j)$$

and

$$s(\hat{w}_1, w_1) < s(\hat{w}_1, w_1)(i < j)$$

Proof. According to the rearrangement inequality:

$$\sum_{i=1}^{N} x_i y_i \leq \sum_{i=1}^{N} x_{\sigma(i)} y_i$$

for every choices of real number $x_1 \leq x_2 ... \leq x_N$ and $y_1 \leq y_2 ... \leq y_N$, and every permutation $\{x_{\sigma(i)}\}$ of $\{x_i(i = 1, ..., N)\}$.

Applying the rearrangement inequality for $x_i = p(w_1, \hat{w_2}^i)$ and $y_i = s(\hat{w_1}, w_1)(i = 0, ..., N)$ we have the minimum of $\sum_{w_2} p(w_1, \hat{w_2}) s(\hat{w_1}, w_1)$. □

To find the optimal adaptation, we calculate over all possible assignments for $ad(\boldsymbol{w})$ and find the minimum $E_{er(\boldsymbol{w})}$ by applying this lemma. In the case of one-value-protection, the steps to find the optimal adaptation in the case of one-value-protection is described in the Algorithm 1.

Algorithm 1

Input: $p(w_1|w_2), s(w_1, w_2)$ for all $w_1, w_2 \in \{1, .., M\}$ and $\hat{w} \in \{1, .., M\}$
Output: Set S of M pairs (w_1, w_2) assigned to adaptor.
procedure
    $min = 1$
    for $i \leftarrow 1$ to M **do**
        $S_t = \emptyset$
        Assign i to w
        Sort $p()$ in increased order to get array $p()$
        Sort $p()$ in increased order to get array $p()$
        Assign w to w.
        $E =$
        if $E < min$ **then**
            $min = E$
            $S = S_t$
    Return S

B. Greedy algorithms for unequal error protection

This section describes an algorithm to find an adaptation to decrease the rate distortion in JSCC. Let $i(w, w_{ad})$ be the indicator function of the adaptation, i.e.:

$$i(w, w_{ad}) = \begin{cases} 1 & \text{if } w \text{ is mapped to } w_{ad} \\ 0 & \text{otherwise} \end{cases}$$

The optimal adaptation can be determined by solving the following problem:

$$\min \sum_{w=1}^{M} \sum_{w_{ad}=1}^{M} p(w) p(\hat{w}_{ad}|w_{ad}) s(w, \hat{w_{ad}}) i(w, w_{ad})$$

subjects to:

$$\sum_{w=1}^{M} i(w, w_{ad}) = 1$$

$$\sum_{w=1}^{M} i(w, w_{ad}) = 1$$

Solving this problem by means of an complete search is impossible since $M!$ cases have to be considered. Instead, we propose this greedy algorithm with fast running time to find a feasible solution. The idea is as follow: from the list of all $w \in M$ in increased order of importance, at each step, we find the most appropriate value of w_{ad} and map it to the most important value from the list. The pseudo code of this greedy algorithm is described in Algorithm 2. The effect of our JSSC with adaptation is estimated by comparing with the traditional UEP strategy [?]. It is clear that the complexity of

Algorithm 2 Greedy algorithm

Input: $p(w), p(\hat{w}_{ad}|w_{ad}), s(w, w_{ad})$ for all $w, w_{ad}, \hat{w}_{ad} \in \{1, .., M\}$
Output: $i(w, w_{ad})$ for all $w, w_{ad} \in \{1, .., M\}$
procedure
  Sort $\{w\}$ in decreased order of importance: $S = \{m_i \in \{1, .., M\}|m_i < m_j \ with \ i < j\}$
  for $i \leftarrow 1$ to M **do**
    for $j \leftarrow 1$ to M **do**
      $i(m_i, m_j) = 0$
  for do $i \leftarrow 1$ to M
    $w_m \leftarrow -1$
    $Sc_{min} \leftarrow \infty$
    $Pe \leftarrow 0$
    for e **do**ach w in T
      $Pe+ = p(w|m_i).s(w, m_i)$
    if then $Pe < Sc_{min}$
      $i(i, w_{min}) = 0$
      $i(w_{min}, i) = 0$
      $i(w, i) = 1$
      $i(i, w) = 1$
      $Pe = Sc_{min}$
      $w_{min} = w$
  Return

this greedy algorithm is $O(M^2)$ where M is the number of source outputs.

IV. Simulation and Results

We have conducted some experiments with and without an adaptation in two scenarios. In the first scenario, we use an adaptation to protect a particular value of compressed speech data in a fixed compression mode. Parameters for the adaptation are calculated by Algorithm 1. In the second scenario, our algorithm minimizes the rate distortion of speech data for all codec modes with the adaptation parameters calculated by Algorithm 2.

A. Simulation description

Our simulation is based on speech data with the Adaptive Multirate (AMR) audio compression [?] as the source code and the traditional convolution code [?] as the channel code. Raw speech data is compressed by the AMR compression to obtain 8 different compression rates. The convolution code with rate 1/3 is used to add redundant data to the compressed

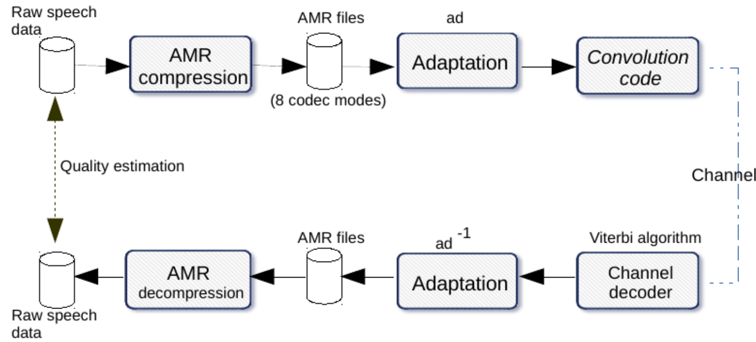

Fig. 2. Comparison of UEP and JSCC with adaptation in one particular AMR codec mode

versions of the speech data. An adaptation was added between the AMR source code and the convolution code. Therefore, the important values of the AMR data are mapped to values which suffered less errors than others when they are transferred over noisy channels. The performance of our proposed method is compared to the traditional UEP strategy [?].

For the sakes of clarity and brevity, we use binary asymetric channel. Other channels will be considered in future works. We use Mean Opinion Score to calculate the quality of speech data after transfer over the channel. Our simulation is described in Fig. 1.

B. Simulation parameter

The blocklength of the channel code input is 8 bits. To estimate the importance of each 8-bit value in an AMR frame, we calculated the probability distribution of channel code input values in a set of 556.300 frames. We remind that an AMR frame includes 4 types of bits with different sensitivities to error: header, class A, class B and class C. We assume that the importance of each channel code input is proportional to its probability distribution. Interestingly, the highest probability distribution falls into the value of the 8-bit header which is the most important part of each AMR frame. In addition, we further assume that in each type, the rate distortion augmentation of 2 source output values is proportional to the Hamming distance between them. Consequently, a coefficient C_{type} is a multiple of the Hamming distance between 2 values of each type to calculate the rate distortion augmentation as follows:

$$s(w_1, w_2) =$$

$$(C_h p_h(w_1) + C_A p_A(w_1) + C_B p_B(w_1) + C_C p_C(w_1)) H(w_1, w_2)$$

where p_h, p_a, p_b, p_c and C_h, C_a, C_b, C_c are the probabilities of each value and the coefficients of header, class A, class B and class C values, respectively. The parameters of the simulation are presented in Table I.

Parameters	Values
Coefficients of bit types	$C_h = 4, C_a = 3, C_b = 2, C_c = 1$
Channel	Binary asymetric channel $p_{00} = 0.9, p_{01} = 0.1$ $p_{10} = 0.2, p_{11} = 0.8,$
AMR modes	4.75, 5.15, 5.9, 6.7, 7.4, 7.95, 10.2 and 12.2 kbits/s
Convolution code rate	1/3
Blocklength	8 bits

C. Performance evaluation

1) Adaptation for one-value protection: This scenario is to protect a specific output value from the source code, when the speech data is compressed with a particular compression rate. Precisely, in AMR compression format, data are divided into frames, each frame has a header and we want to protect the 8-bit-header value of every frame in an AMR codec. The performance of the adaptation for the first scenario is shows in Fig. 2.

As seen in the figure, while the traditional method is ineffective for transfer of data over a noisy channel, by using different adaptations on different scenarios, protection for header values of every AMR frames attains considerably high quality of speech each mode, with 6 of 7 received codec modes having MOS greater than 2.5. However, we can apply these adaptations only in the case of knowing what particular AMR codec mode is transferred. The performance of adaptation given by the Algorithm 2 is estimated for the second scenario when the codec mode is not specified.

2) Adaptation for general case: For models in which the source and channel code are performed separately and independently, Fig. 3 represents a comparison of our JSCC with the adaptation model and the traditional UEP model. It was apparent beforehand that the quality of files encoded by lower bit-rates is smaller than that of files with greater bit-rates. In Fig. 3, one can observe this standard trend of the speech quality attained by different AMR compression rates. In both experiments with the traditional strategy and the JSCC with adaptation, speech quality quantified by MOS is increased over all the AMR codec mode from 4.75 to 10.02. In traditional method, although AMR data are protected by channel code, the presence of channel errors leads to disastrous quality: MOS less than 2.0 in 6 codec modes, which is a level of speech quality that is hard for human ears to distinguish. In contrast, as can be seen in the figure, the JSCC system equipped with an appropriate adaptation attained considerably higher quality speech data than the traditional model, with 6 out of 7 received codec modes having MOS greater than 2.0.

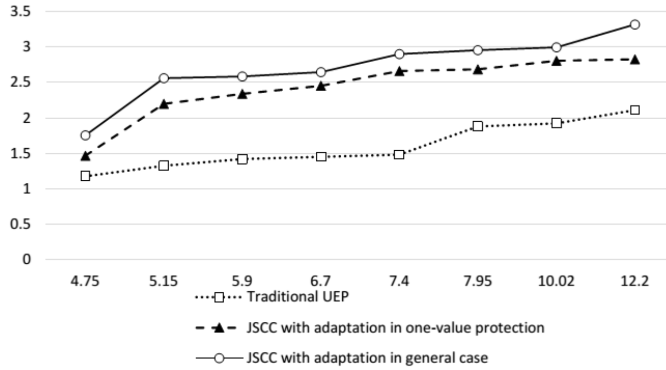

Fig. 3. Comparison of UEP and JSCC with adaptation in one particular AMR codec mode

V. Conclusion

To improve the quality of data transmission in JSCC strategy, we have shown a new method to exploit the unequal importance of source outputs and the nature of noisy channels. As evidenced by the numerical results, the adaptation, which applies to map the source outputs to appropriate channel code inputs, can outperform the traditional UEP method to transmit speech data over noisy channel. Unless the messages has the same effect that caused by the channels while they are transferred, our JSCC with adaptation offers significant advantage in the finite blocklength mode.

References

[1] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.

[2] V. Kostina and S. Verdú, "Lossy joint source-channel coding in the finite blocklength regime," *Information Theory, IEEE Transactions on*, vol. 59, no. 5, pp. 2545–2575, 2013.

[3] V. Chande and N. Farvardin, "Progressive transmission of images over memoryless noisy channels," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 6, pp. 850–860, 2000.

[4] V. Stanković, R. Hamzaoui, and D. Saupe, "Fast algorithm for rate-based optimal error protection of embedded codes," *Communications, IEEE Transactions on*, vol. 51, no. 11, pp. 1788–1795, 2003.

[5] P. G. Sherwood, X. Tian, and K. Zeger, "Channel code blocklength and rate optimization for progressive image transmission," in *Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE*. IEEE, 1999, pp. 978–982.

[6] R. Jain, K. K. Ramakrishnan, and D. M. Chiu, "Effect of eep and uep on channel coding for amr," Nokia, Hannover, Germany, Tech. Rep. TSG-RAN WG1-7.

[7] O. Y. Bursalioglu and G. Caire, "Is unequal error protection useful?" in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*. IEEE, 2011, pp. 1402–1406.

[8] 3GPP, "3gpp specification detail: Ansi-c code for the floating point adaptive multi-rate (amr) speech codec," Nokia, Tech. Rep., 2014.

[9] A. J. Viterbi, "An intuitive justification and a simplified implementation of the map decoder for convolutional codes," *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 2, pp. 260–264, 1998.