EVALUATION OF PACKET ERROR RATE IN WIRELESS NETWORKS

Ramin Khalili, Kavé Salamatian
LIP6-CNRS, Université Pierre et Marie Curie.
Paris, France.
Ramin.khalili, kave.salamatian@lip6.fr

Abstract

Bit Error Rate (BER) and Packet Error Rate (PER) are important Quality of Service Parameters for Wireless network. Most of researches in QoS have been devoted to the analysis of BER which gives insight to the mean behavior of the wireless network. However the mean behavior is not sufficient in a lot of scenarios and more precise characterization of the error process is needed. An important example where the mean behavior is not sufficient is PER evaluation. As residual errors at the output of physical layer are not uniformly distributed, the distribution of these error events is important for deriving PER. Not taking into account this distribution and supposing for example that errors are uniformly distributed as done in the a large proportion of the published paper in wireless networking lead to gross overestimation of PER that can goes to tenfold factors.

This paper presents a new analytic approach to derivate the Packet Error rate in Wireless networks where convolutional codes are used jointly with Viterbi decoder. It is based on a precise analysis of the error process at the output of a Viterbi decoder. Some asymptotic bound based on error exponent are derived and it is shown that these bound are sufficiently tight to be applied in practice. Based on this analysis a closed form formulas approximation for derivation of PER in AWGN is proposed. It is shown that these closed form formulas give very tight upper bound for the PER.

Keywords: Qos, Packet error rate, Convolutional code.

1. Introduction

Packet Error Rate (PER) is an important Quality of Service Parameters for Wireless network. However it is mostly derived using BER and an error independence assumption. Even if BER gives important insight to the mean behavior of a wireless network, however it is not sufficient for PER derivation. For this purpose a more precise characterization of the error process is needed as residual errors at the output of physical layer are not uniformly distributed (because of error correcting codes). Till now PER was mainly derived based on a uniform error distribution hypothesis that lead, as we will show in the rest of the paper, to large overestimation of PER, with proportion that can goes to a tenfold proportion.

However analysis of error event distribution is not easy because of the peculiar interaction of the particular error correcting codes used in the physical layer of wireless networks. This paper addresses the analysis of the distribution of error events at the output of Viterbi decoders that are widely used for decoding convolutional codes. This distribution is classically analyzed using the weight distribution of the code. This distribution can be obtained using algorithms based on state transition matrix inversion and properties of the channel [1]. However the complexity involved in this derivation can be quickly tremendous specially when the memory of the code goes larger and we have to study long error events. Moreover this calculation is specific to each particular code and does not give a parametric formulation that might be used in optimization of system parameters.

Our goal in this paper is to derive an accurate closed form formula for predicting PER that can be used in practice for designing systems. As presented in the complete survey of error modeling schemes for channels in wireless communications [8]three main approaches have been applied to deriving PER in wireless channels. In the literature most of models have assumed the error process to be *i.i.d.* As we will see in the forthcoming this lead to rough overestimation. The second type of approaches have described the error process as Markov model (for example the well-known Elliot-Gilbert model) [9]but these description failed in providing closed form formula based on code parameters that will enable to design communication systems. The third type of approach was empirical distribution based modeling. Whenever this approach lead to realistic models but it cannot be really applied in designing wireless systems as changing the parameters of a system will change its empirical behavior. The approach presented in this paper leads to a relatively simple formula helpful for choosing the parameters of a system. Up to our knowledge, no formula with the achieved accuracy have never been published in the litterature.

The approach developed in this paper uses an asymptotic technique based on error exponent. It is shown that despite the general belief that asymptotic bounds do not give applicable results in practice, they are very useful for the analysis of the distribution of error event lengths at the output of Viterbi decoding of convolutional codes. The idea is based on the code termination technique first presented in [4], that relates error probability of a random convolutional code to the error probability of a particular block code. This technique enables us to bound the probability of long error events. Obtained error event length distribution used jointly with the distribution of errorless period lead to a simple model for describing residual error at the output of the Viterbi decoder applying soft decoding. This model can be easily used in the context of wireless link simulation for generating residual errors. It can also be used to precisely derive the PER at the output of the Viterbi decoder.

In the next section error event error exponent analysis are defined and concepts needed in the paper are introduced. The section 3 will present the code termination idea and give the basic bound on the distribution of error event length. The analysis of the error-less period and a closed form formula for PER calculation will be presented in this section. The experimental validation will be shown in section 4.

2. Concepts

The performance of Convolutional codes is analyzed by following the path chosen by the Viterbi algorithm in the code trellis [1]. An error occurs if a divergence between the path chosen by Viterbi algorithm and the intial path occurs. An error event is defined as any divergence of the decoded path at the receiver from the initially followed path in the encoder.

The three following simple rules can be applied to parse the error sequence into error events and error free periods: A received symbol is inside an error event or is inside an error free period. Each error event begins with an erroneous received symbol and each error event ends by a sequence of ν bits received without any errors, where ν is the memory of the code.

One of our objectives in this paper is to derive the distribution of the number of decoded bits that are inside an error event. We will call this distribution Error Event Length (EEL) distribution. It is clear that all bits inside error events are not erroneous, nevertheless they should be considered as highly unreliable. Analysis of error event is much simpler that the study of errors themselves as the derivation is not polluted by specific code structure. Globally the shape of the distribution of error event length seems to be generic between codes sharing same basic characteristics as rate, memory and free distance d_{free}. We have shown in Fig. 1 the EEL distribution as observed in simulation at the output

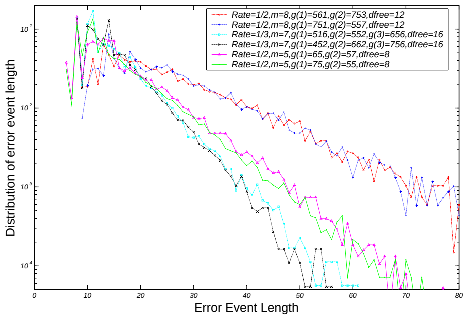


Figure 1. EEL distribution with SNR=2 dB for 6 different codes

of an AWGN with a SNR equal to 2 dB for different codes (see section 1.4 for details). It can be observed that codes sharing the same characteristics have similar EEL distribution regardless of the fact that their structure and even BER can be very different. Asymptotic analysis of error correcting block codes through error exponent is a classical approach in information theory [10]. This approach is interesting as it gives closed form bounds on the error behavior of codes. These bounds can be readily used to design and optimize error correcting schemes. For long codes, the error probability of block codes and convolutional codes is upper-bounded in terms of error exponent. The error exponent $E(R)$ is defined as $E(R) = -\frac{1}{N}\log P_e(N,R)$ where $P_e(N,R)$ is the error probability of a block code of length N and rate R.

There are classically two kinds of bounds: lower bound given by the sphere packing bound and upper bound given by the random coding bound. The error behavior of a practical code should be somewhere between these two bounds. As we are interested in upper bound for code behavior we are more interested in random coding bound and error exponents. Random coding error exponents for soft decoding over Additive White Gaussian Noise (AWGN) channel might be derived using bounds first obtained by Shannon itself in [3]. Because of space limitation and technicalities we are not going on the details of how to derive these error exponents. More detail about this topic can be find in [7]and [2].

3. Error Event Length distribution and PER in convolutional codes

This section introduces code termination technique [4]that relates convolutional codes and block code performances. The error event and errorless period analysis and packet error rate derivation results from this idea.

The code termination idea consists of terminating a convolutional code after a certain number, let say τ, of input information symbols. The termination is made by forcing the input to zero for at least ν input symbol until the encoder has returned to the null state. This termination creates a block code that can be analyzed using block code error exponents. The performance of a convolutional code is analyzed by considering a sequence of block codes of increasing length and rate constructed using the code termination technique.

Let's $\mathcal{C}$ be a (n, k, ν) convolutional code with constraint length ν and rate $r \stackrel{\Delta}{=} \frac{k}{n}$. Terminating a convolutional code after τ time units will lead to a block code with length $N = n\tau$, $M_0 = 2^{(\tau-\nu)k}$ codewords and an overall rate $R \stackrel{\Delta}{=} \frac{1}{N}\log_2 M_0 = r(1-\theta)$, where $\theta = \frac{\nu}{\tau}$ is the ratio of the code constraint length to block length.

Theorem 1 (Error event length distribution [4]) *Using the code termination technique the probability per unit time of an error event of length τ time units in an (n, k, ν) convolutional code is bounded by :*

$$\mathbb{P}rob\{\tau, \mathcal{E}\} \quad \leq e^{-n\tau E((1-\frac{\nu}{\tau})r)} = e^{-n\nu E((1-\theta)r)/\theta}$$

where $\theta = \frac{\nu}{\tau} < 1$, $r = \frac{k}{n}$ is the convolutional code rate and $E(.)$ is the block code error exponent One can remark that this result is the same as for a block code of length $n\tau$ and rate $R = (1-\theta)r$ $\square$.

This theorem gives the distribution of error event length.

However the output of a Viterbi decoder contains error events mixed with errorless periods. The characterization of the output of the Viterbi remains incomplete without describing errorless periods. In this widely believed that errorless periods length and error event length are independent. However a more precise analysis of the Viterbi algorithm shows that this assumption is not correct, specially for low SNR. We define a decoding epoch as the interval between the end of two consecutive error events. A decoding epoch consists of an errorless period followed by an error event. In fact the length of an error event and an error epoch can be assumed as independent. We will argue in the forthcoming that the distribution of error epochs can be easily derived using a geometric memoryless distribution. The distribution of errorless event is derived based on these two distributions.

Lets α_i be the length of the i^{th} decoding epoch, and β_i being the length of the i^{th} error event. The value $\gamma_i = \alpha_i - \beta_i$ gives the length of the errorless period preceding the i^{th} error event. Because of the memoryless property of the AWGN channel error epochs are independent and memoryless events. Moreover the probability that a decoding epoch finishes at any bit j can be easily obtained through a classical derivation that can be found in basic coding textbooks [1, 4]. Let's $EER(SNR)$ (Error Event Rate as a function of SNR) be this probability for a given SNR over the AWGN channel.

$$EER(SNR) \approx A_{d_{free}} e^{R \cdot SNR \cdot d_{free}}$$

where $A_{d_{free}}$ is the number of path of length d_{free} in the code trellis and R is the code rate. As the decoding epoch last at least ν bits, $\alpha_i = \nu + \alpha$ where α follows a Bernoulli process with parameter $EER(SNR)$. The mean length of decoding epochs is therefore equal to $\mathbb{E}\{\alpha_i\} = \nu + \frac{1}{EER(SNR)}$. The characterization of error and error epoch events enables the development of a simulator of the error process at the output of the Viterbi decoder.

Derivation of PER need the distribution of errorless period length. This distribution can be obtained precisely using the fact that the error event length and decoding epoch length are independent as a convolution:

$$f_\gamma = f_\alpha * f_{-\beta}$$

However this formulation does not give a close form. For high value of SNR a decoding epoch consists of a long errorless period followed by a small error event. In this situation we will approximate the distribution of errorless period length with Bernoulli process with mean length that can be derived using the mean length of decoding epoch and error event length. This assumption will be shown to be valid for the range of SNR that have practical relevance.

3.1 Packet Error Rate derivation

We have shown previously that error exponent are useful for derivation of distribution of error event length. In the forthcoming we will use an approximation of these exponent to derive a closed form formula for PER derivation. As shown in [3], for large SNR the error exponent of a block code can be approximated by using only the linear part of the random coding error exponent. While this is a rough estimate of error exponent, we will show that it is tight enough to give tight upper bound for PER. This approximation gives the error exponent for a block code (n_b, k_b) of rate $r_b = \frac{k_b}{n_b}$ with a Signal to Noise Ratio given by s as :

$$E(s, r_b) = s/2 - \sqrt{2sr_b} + r_b \tag{1}$$

Using theorem 1, the probability of an error event of length τ in a convolutional code (k_c, n_c, ν) of rate r_c and constraint length ν is equivalent to the error probability of a block code with rate $r_b = (1-\nu/\tau)r_c$ giving based on the previous linear approximation:

$$\begin{aligned}\mathbb{P}\text{rob}\{\tau, s\} &\leq \exp\left[-n_c \tau E\left(s, (1-\tfrac{\nu}{\tau})\right) r_c\right] \\ &\approx \exp[-n_c\tau(s/2 - \sqrt{2 \cdot s(1-\tfrac{\nu}{\tau})r_c} + (1-\tfrac{\nu}{\tau})r_c)]\end{aligned}$$

Moreover, the average error event length at the output of Viterbi decoder can be obtained as below:

$$\overline{\tau(s)} = \frac{\sum_{\tau=\nu+1}^{\infty} \tau \cdot Prob(\tau, s)}{\sum_{\tau=\nu+1}^{\infty} Prob(\tau, s)}$$

The average error event length can be approximated as (see [2]) :

$$\overline{\tau(s)} = (\nu+1) + \frac{1}{n_c(s/2 - \sqrt{2sr_c} + r_c)}$$

As the mean length of decoding epochs $(\overline{L})$ is known to be equal to :

$$\overline{L(s)} = \frac{1}{EER(s)}$$

for SNR=s, the mean length of errorless period $\overline{\omega(s)}$ can be calculated as :

$$\overline{\omega} = \overline{L} - \overline{\tau(s)} = \frac{1}{EER(s)} - (\nu+1) + \frac{1}{n_c(s/2 - \sqrt{2sr_c} + r_c)}$$

However as explained before, for high SNR it is possible to assume that errorless period length follows a geometric distribution with parameter $\lambda(s)$ which is related to the mean length $\overline{\omega}$ through :

$$\lambda(s) = \frac{1}{\overline{\omega}} = \frac{EER(s)}{1 - [(\nu+1) + \frac{1}{n_c(s/2-\sqrt{2sr_c}+r_c)}]EER(s)} \tag{2}$$

Lets suppose that the convolutional code decoder is reset at the begin of each received packet. The probability that a packet contain an error event (or a part of an error event) is simply given by the probability that the errorless period begin at the first bit of a newly received packet lasts for less than the packet length N. In other term we have :

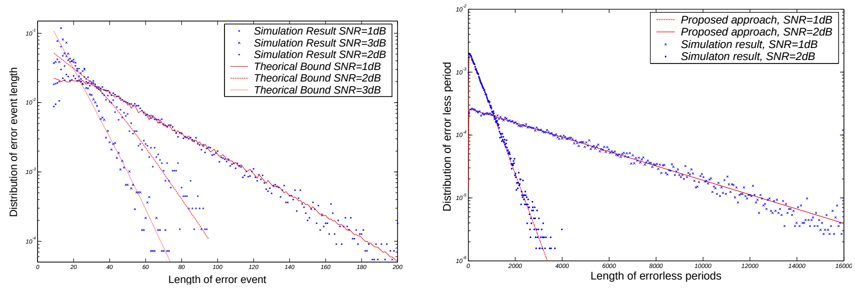
$$PER = \sum_{i=0}^{N-1} \lambda(s)[1-\lambda(s)]^i = 1 - (1-\lambda(s))^N \tag{3}$$

where $\lambda(s)$ is given in Eq. 2 and EER is given by equation 1.3. All the parameters of the new formula can be easily obtained using basic hypothesis and we will show in the experimental validation part that this formula gives remarkably tight bounds for PER. It is noteworthy that the obtained relation have the same form as the formula obtained under *i.i.d* assumption however the value of the parameter are different for *i.i.d* case and for the case under study.

4. Experimental validation of EEL distribution

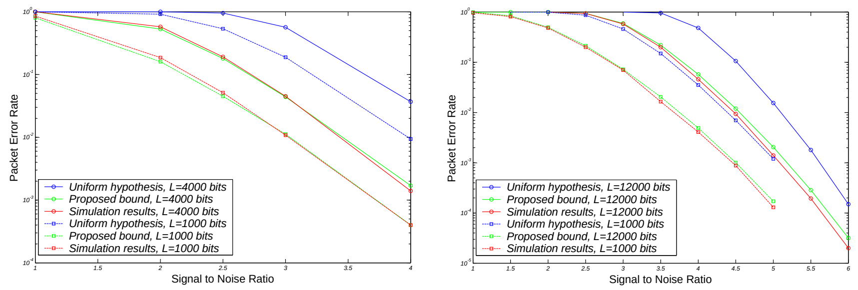
In this section we will validate the proposed approach for deriving EEL distribution by comparing empirically derived EEL distribution and the upper bounds given by the previous analysis. We illustrate the approach with two codes. The first convolutional code has been defined as a standard for the UMTS physical layer [6]and has rate $r = 1/2$ with constraint length 9 and generator polynomial in octal format defined as (561,753). This code has a relative large memory that makes difficult the application of the classical weight distribution approach. The second code is the convolutional code used in IEEE 802.11a physical layer. This code has a rate $r = \frac{1}{2}$, constraint length $\nu = 6$ and generator polynomial (133, 171). We have simulated a communication systems consisting in a convolutional encoder followed by a BPSK modulator. The encoded modulated signal is sent over an AWGN channel, and decoded at the receiver by a soft decoding Viterbi algorithm without memory truncation. Each simulation tests have been applied over 80 Mbits of data.

Applying the rules defined in section 1.2, an error event is detected if an error occurs after a sequence of at least ν bits without error and is terminated after a sequence of ν bits without errors. We have compared the error event length distribution observed with the error event length predicted following the method developed in section 1.3. The comparison is shown in fig. 1.4 for an AWGN channels with SNR=1.0 dB, 2.0 dB and 3.0 dB. Figure 1.4 shows that globally the upper bound analysis provided here is able to predict the shape of the error event length distribution. We have also apply the same procedure for other convolutional codes, the results are similar. Another important distribution introduced in previous section is the errorless event distribution. We show in Fig. 1.4 the comparison of the errorless period length observed over simulation data and the value predicted using the proposed method. One more time the figure shows that the global shape is predicted. However we have a large dispersion around the predicted curve distribution. Moreover the curve shows a nice adequation with an exponential behaviour of the Errorless period. This validates somewhat the



(a) ***Error event length distribution*** **(b)** ***Errorless period length distribution***

Figure 2. EEL and ELL distribution for the UMTS (561,753) code with memory constraint $\nu = 9$ over an AWGN channel with SNR=3.0dB, 2.0dB and 1.0dB



(a) ***UMTS (561,753) code with*** $\nu = 9$ ***for two packet sizes L=1000 bits and 4000 bits*** **(b)** ***WIFI (171,133) code with*** $\nu = 7$ ***for two packet sizes L=1000 bits and 12000 bits(= 1500 bytes)***

Figure 3. PER of convolutional code over an AWGN channel as a function of SNR

assumption we made previously for deriving the PER about the errorless period being distributed exponentially. The observed variations in the tail come from the insufficient number of samples available.

We have used the same simulated set of data to derive PER for different SNR over an AWGN channel. The obtained value are compared with the prediction given by the formula given in Eq. 3, we have also compared the value with what have been predicted using the uniform hypothesis (where $PER = 1 - (1 - BER)^N$) that is largely used in networking community. The results are shown in Fig. 3(a) for the UMTS code and in Fig. 3(b). For the WIFI case we have given the result for the 1500 bytes (=12 000 bits) which is maximal size of packet send over WIFI channel. As can be seen from these two figures the uniform error hypothesis gives results that are very far from realistic values. The presented curves show that the classical approach lead to an over-estimation of PER with a factor of almost 10! However in the same situation the proposed formula lead to much tighter bound(overestimation by a factor of 1.8 at most).

5. Conclusion

This paper has presented a new approach derivation of error event lengths at the output of Viterbi decoder for convolutional codes. The approach is based on asymptotic error exponents. We have also derived the distribution of errorless period length. This last derivation enables us to obtain a precise and tight characterization of Packet Error Rate at the output of Viterbi decoders. A software implementation of the error distribution model has been developed and is available for downloading at `http://www-rp.lip6.fr/salamat`.

References

[1] S. Lin and D. Costello, " Error Correcting Coding: Fundamentals and Applications ", Prentice Hall, 1983.

[2] R. Khalili and K. Salamatian, " A new analytic approach to evaluation of Packet Error Rate in Wireless Networks ", Technical raport, LIP6-PERmodel, 2004.

[3] C.E. Shannon, " Probability of Error for Optimal Codes in a Gaussian Channel ", Bell System Technical Journal, 1959, vol. 38.

[4] G. D. Forney Jr., " Convolutional codes II. Maximum-likelihood decoding ", Information and Control, 1974, vol.25, numb.3, 407-421.

[5] J. Justesen and J. D. Andersen, " Critical Length of Error Events in Convolutional Codes ", IEEE Trans. on Inf. theory, July 1998.

[6] ETSI, " Universal Mobile Telecommunications System (UMTS); Multiplexing and channel coding (TDD) ", 3GPP TS 25.222 ver. 5.2.1 Rel. 5, October 2002.

[7] S. Yousefi and A. K. Khandani, " A new upper bound on the maximum likelihood decoding error probability of linear binary block codes in AWGN interference ", IEEE Trans. on Information theory, 2003.

[8] H. Bai, H. Aerospace and M. Atiquizzaman, "Error modeling scheme for fading channels in wireless communications: A survey", IEEE Com. Surveys, vol. 5, No.2, Forth quarter 2003.

[9] M. Zorzi, R. R. Rao, L. B. Milstein, "On the accuracy of a first-order Markov model for data block transmission on fading channels", IEEEICUP'95, pp.211-215, Nov. 1995.

[10] R. G. Gallager, "Information Theory and reliable Communication," Wiley, New York, 1968.