Viterbi Decoding for Convolutional Code over Class A Noise Channel

Yuki NAKANO†, Daisuke UMEHARA‡, Makoto KAWAI‡, and Yoshiteru MORIHIRO‡

†School of Electrical and Electronic Engineering
‡Graduate School of Informatics
Kyoto University
Yoshidahonmachi, Sakyoku, Kyoto, 606-8501, Japan
Phone: +81-75-753-5960, Fax: +81-75-753-5349
E-mail: yuki@forest.kuee.kyoto-u.ac.jp, dumehara,kawaki,morihiro}i.kyoto-u.ac.jp

Abstract

Power line channel often suffers from impulsive interferences generated by electrical appliances. Therefore, power line communication (PLC) suffers the degradation due to such the impulsive interferences. Middleton’s Class A noise model is frequently utilized for the modeling of such the impulsive noise environment.

In this paper, we deal with the Viterbi decoding for convolutional code over additive white Class A noise (AWAN) channel. We propose a Viterbi decoder for convolutional code optimized in AWAN channel. In addition, we show the performance of the proposed Viterbi decoder for convolutional code in Class A noise environment by computer simulation.

1. Introduction

Residential power line is one of the most attractive communication media for home networking, since almost all rooms in a house have its outlets. However, many electrical appliances frequently cause man-made electromagnetic noises on power line channel. Such the man-made noises have the impulsive feature, and then are one of technical problems for the realization of the power line communications with high rate and high reliability. We introduce Middleton’s Class A noise model into a statistical model for such the impulsive noises. Class A noise channel proposed by Middleton [1] is one of non-Gaussian noise channels and is currently applied to the modeling of man-made impulsive noise channels, for example, wireless channel, power line channel and so on.

The statistical characteristic of Class A noise is much different from that of Gaussian noise. Therefore, the conventional decoders optimized in additive white Gaussian noise (AWGN) channel, in general, are not suitable for Class A noise environment. A variety of optimum receivers for Class A noise have been discussed [2–10]. On the assumption that there are many independently identically distributed (iid) Class A noise samples in one symbol duration, i.e. the noise bandwidth is wider than the signal bandwidth, the optimum receivers for single carrier modulations [2–4] and trellis coded modulation (TCM) [5, 6] have been developed, and then it has been shown that their bit error rates (BERs) are lower than those of the receivers optimized in Gaussian noise. On the other hand, Häring et. al have proposed an effective iterative detection using MMSE (minimum mean square error) estimator for complex number codes including OFDM (orthogonal frequency division multiplexing) and PSS (parallel transmitting spread spectrum) [7, 8]. Also Umehara et. al have shown an iterative detection using normalized LOBD (locally optimum Bayes detector) for PCSS (parallel combinatory spread spectrum) system [9, 10]. The performances of these iterative decodings have a remarkable improvement as compared with those of the conventional decodings on the assumption that signal bandwidth is equal to noise one. However, Viterbi decoding for convolutional code, optimized in Class A noise, has not been discussed on the assumption that signal bandwidth is equal to noise one.

In this paper, we propose an optimum Viterbi decoder for convolutional code over AWAN channel. One of the advantages of our proposed decoder is that there is no need to configure the conventional Viterbi decoder. Our proposed decoder can be established only by putting a preprocessing device on the front-end of the conventional Viterbi decoder. Then, we evaluate the BER performance of our proposed decoder over AWAN channel. Furthermore, we compare the performance of our proposed decoder with that of the conventional Viterbi decoder over AWAN channel by computer simulation.

In Section 2, we present the Middleton’s Class A noise model and a simplified Class A noise model [4]. Also, we explain the convolutional code and the Viterbi decoding for the convolutional code optimized in AWGN channel. In Section 3, we propose a Viterbi decoding for convolutional
code over AWAN channel. In Section 4, we show the performance of the proposed decoder over AWAN channel by computer simulation. In Section 5, we present the conclusions.

2. Preliminaries

2.1. Impulsive noise model

2.1.1. Class A noise model

Power line channel is different from many other communication channels. One of characteristics of power line channel is impulsive interferences generated from electrical appliances connected to power line. Such the impulsive noise is one of the serious factors influencing digital communications over power line channel. The occurrence of the impulsive noise may cause bit or burst errors in data transmission. Middleton’s Class A noise model is one of models for impulsive noise environments. The probability density function (PDF) of Class A noise is given by

\[
P_{A,G,\sigma^2}(z) = \sum_{m=0}^{\infty} \frac{e^{-A} A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m}} \exp \left( -\frac{z^2}{2\sigma_m^2} \right),
\]

with

\[
\sigma_m^2 := \frac{\sigma^2_m A + \Gamma}{1 + \Gamma},
\]

where \( A \) is the impulsive index, \( \Gamma := \sigma_0^2/\sigma_1^2 \) is the GIR (Gaussian-to-impulsive noise power ratio) with Gaussian noise power \( \sigma_0^2 \) and impulsive noise power \( \sigma_1^2 \), and \( \sigma^2 = \sigma_0^2 + \sigma_1^2 \) is the total noise power. The noise \( z \) followed by Eq. (1) always includes the background Gaussian noise with power \( \sigma_0^2 \). On the other hand, sources of impulsive noise are distributed with Poisson distribution \( (e^{-A} A^m)/m! \). If one impulsive noise source generates a noise, then the noise is characterized by the Gaussian PDF with variance \( \sigma_1^2 / A \). Consequently, at a certain observation time, assuming that the number of impulsive noise sources is \( m \) which is characterized by Poisson distribution with mean \( A \), the noise of the receiver is characterized by the Gaussian PDF with variance \( \sigma_m^2 = \sigma_0^2 + (m\sigma_1^2 / A) \). The larger \( A \), the impulsive noise will be continuous, and then the Class A noise is led to be more likely to the Gaussian noise. In particular, if \( A \) is nearly equal to 10, the statistical feature of the Class A noise is almost similar to that of the Gaussian noise [1].

2.1.2. Simplified Class A noise model

As the PDF of Class A noise consists of infinite series, it is intractable to develop receivers optimized in Class A noise. In [4], Kusao et al. have simplified the PDF of Class A noise. We utilize this simplified Class A noise model. If impulsive index \( A \) is smaller than 0.25, Eq. (1) is approximate to the sum of three terms in \( m = 0, 1, 2, \)

\[
\hat{P}_{A,G,\sigma^2}(z) = \frac{e^{-A}}{\sqrt{2\pi}} \left( \frac{1}{\sigma_0} e^{-z^2/2\sigma_0^2} + \frac{A}{\sigma_1} e^{-z^2/2\sigma_1^2} + \frac{A^2}{2\sigma_2} e^{-z^2/2\sigma_2^2} \right),
\]

with

\[
\sigma_0^2 = \sigma^2 \frac{\Gamma}{1 + \Gamma}, \quad \sigma_1^2 = \sigma^2 \frac{1/A + \Gamma}{1 + \Gamma}, \quad \sigma_2^2 = \sigma^2 \frac{2/A + \Gamma}{1 + \Gamma}.
\]

Further, Eq. (2) can be approximate to

\[
\hat{P}_{A,G,\sigma^2}(z) = \max_{m=0,1,2} \left[ \frac{e^{-A} A^m}{m!} \cdot \frac{1}{\sqrt{2\pi\sigma_m}} e^{-z^2/2\sigma_m^2} \right] = \begin{cases} 
\frac{e^{-A}}{\sqrt{2\pi\sigma_0}} e^{-z^2/2\sigma_0^2}, & 0 \leq |z| < a, \\
\frac{e^{-A}}{\sqrt{2\pi\sigma_1}} e^{-z^2/2\sigma_1^2}, & a \leq |z| < b, \\
\frac{e^{-A}}{\sqrt{2\pi\sigma_2}} e^{-z^2/2\sigma_2^2}, & b \leq |z|,
\end{cases}
\]

with

\[
a = \sqrt{\frac{2\sigma_0^2\sigma_1^2}{\sigma_0^2 - \sigma_1^2} \ln \left( \frac{\sigma_0 A}{\sigma_1} \right)}, \quad b = \sqrt{\frac{2\sigma_1^2\sigma_2^2}{\sigma_1^2 - \sigma_2^2} \ln \left( \frac{\sigma_1 A}{\sigma_2} \right)}.
\]

As impulsive index \( A \) becomes smaller, Eq. (1) sufficiently approximates Eq. (3). We call Eq. (3) a simplified Class A noise model. Some receivers optimized in the simplified Class A noise model have been proposed [4–6].

2.2. Convolutional code

Error-correcting code is classified into two kinds. One is block code, the other is trellis code. Convolutional code belongs to trellis code. In convolutional code, each \( k \)-bit information block (\( k \)-bit string) is encoded to an \( n \)-bit block, where \( n \) is larger than \( k \), and the current code block is related to the last \( L \) information blocks. In such the case, the code rate is defined by \( k/n \). The code rate \( k/n \) has not only the meaning of rate but also the meaning that the numerator \( k \) is the information block length and the denominator \( n \) is the code block length. Therefore, code rate \( 2/4 \) is different from code rate \( 1/2 \). The \( L \) is the number of code blocks with which a information block relates directly in encoding and is called a constraint length.
By using the delay operator $D$, the information series $(m_0m_1m_2\cdots)$ and the code series $(w_0w_1w_2\cdots)$ is expressed by

\[ M(D) = m_0 + m_1D + m_2D^2 + \cdots, \]
\[ W(D) = w_0 + w_1D + w_2D^2 + \cdots, \]

where information series $(m_0m_1m_2\cdots)$ is each $k$-bit information block and code series $(w_0w_1w_2\cdots)$ is each $n$-bit code block. The $i$-th component $M_i(D)$ ($i = 1, 2, \cdots, k$) of $M(D) = (M_1(D), M_2(D), \cdots, M_k(D))$ is given by

\[ M_i(D) = m_{i,0} + m_{i,1}D + m_{i,2}D^2 + \cdots, \]

and the $i$-th component $W_i(D)$ ($i = 1, 2, \cdots, n$) of $W(D) = (W_1(D), W_2(D), \cdots, W_n(D))$ is given by

\[ W_i(D) = w_{i,0} + w_{i,1}D + w_{i,2}D^2 + \cdots. \]

The code series $W(D)$ is expressed by

\[ W(D) = M(D)G(D), \]
\[ [W_1(D) \cdots W_n(D)] = [M_1(D) \cdots M_k(D)]G(D), \]

where $G(D)$ is a $k \times n$ matrix whose elements are polynomials of $D$, and is expressed by

\[ G(D) = \begin{bmatrix} G_{1,1}(D) & G_{1,2}(D) & \cdots & G_{1,n}(D) \\ G_{2,1}(D) & G_{2,2}(D) & \cdots & G_{2,n}(D) \\ \vdots & \vdots & \ddots & \vdots \\ G_{k,1}(D) & G_{k,2}(D) & \cdots & G_{k,n}(D) \end{bmatrix}. \]

The $G(D)$ is called a generator matrix of convolutional code, and $G_{i,j}(D)$, which is $(i,j)$-th element of $G(D)$, is called a generator polynomial.

We utilize the convolutional encoder with code rate $1/2$ and constraint length 3, whose generator matrix is

\[ G(D) = \begin{bmatrix} 1 + D^2 & 1 + D + D^2 \end{bmatrix}, \]

in our computer simulation. Figure 1 illustrates the block diagram of the convolutional encoder. The block of $D$ represents a shift register and the block of $\oplus$ represents an XOR.

2.3. Viterbi decoding

The Viterbi decoding is the most general decoding for convolutional code by using an asymptotically optimum decoding technique. The Viterbi decoding is an efficient and recursive algorithm that performs the maximum likelihood (ML) decoding. A noisy channel causes bit errors at the receiver. The Viterbi algorithm finds the most likely bit series that is closest to the actual received series. The Viterbi decoder uses the redundancy, which is imposed by the convolutional encoder, to decode the bit stream and correct the errors. At first, we explain the various terms with respect to Viterbi decoding.

- **Trellis diagram**
  The trellis diagram is the diagram representing a transition of the state of shift registers in convolutional encoder by taking time axis to the horizontal axis.

- **Traceback depth**
  The traceback depth is the number of trellis states processed before the Viterbi decoder makes a decision on a bit. For blocks of data, the best performance is achieved if decoding decisions are delayed until all input symbols have been processed. For continuous streams, this is not possible, and there is no benefit in increasing traceback depth beyond several times constraint length. Empirically, it is known that traceback depth of about $5 \sim 6$ times length of constraint length in convolutional encoder over AWGN channel is useful.

- **Hard/soft decision**
  In hard decision, the receiver deliver a hard symbol equivalent to a binary $\pm 1$ to the Viterbi decoder. In soft decision, the receiver deliver a soft symbol multi-leveled to represent the confidence in the bit being positive or negative to the Viterbi decoder.

We assume that code series of convolutional code with code rate $k/n$ is transmitted over memoryless stationary channel and series $Y = y_0y_1y_2\cdots y_{N-1}$ is received, where $N$ is arbitrarily large positive integer. The $t$-th block of received series is $y_t = (y_{1,t}, y_{2,t}, \cdots, y_{n,t})$, and $y_{i,t}$ is output symbol of channel. The ML decoding outputs the code series $W = (w_0w_1w_2\cdots w_{N-1})$ whose likelihood function $P(Y|W)$, which is the conditional probability of $Y$ given $W$, is maximal over all possible code series. As channel is memoryless stationary, this likelihood function is written by

\[ P(Y|W) = \prod_{t=0}^{N-1} P(y_t|w_t). \]
Moreover, this logarithm is represented by

$$\ln P(Y|W) = \sum_{t=0}^{N-1} \ln P(y_t|w_t).$$

The ML decoding can be established by calculating the logarithm $\ln P(y_t|w_t)$ for any $t$ and code block $w_t$, summing $\ln P(y_t|w_t)$ for all $t$, and picking up the code series corresponding to the maximal sum. The $\ln P(y_t|w_t)$ for code block $w_t$ corresponding to the branch of trellis diagram is called a branch metric. In general, the total number of code series $W$ is $2^{nN}$, where $nN$ is the length of the received series $Y$. If $nN$ is large, the ML decoding may require the enormous calculations. However, we can reduce the number of series in the trellis search by using the Viterbi decoding. The Viterbi decoding is a sequential trellis search algorithm for performing the ML decoding.

We explain the Viterbi decoding over AWGN channel. The probability density function (PDF) of the Gaussian noise is expressed by

$$P_{g2}(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left( -\frac{z^2}{2\sigma^2} \right),$$

where $z$ is a random variable of noise. By using transmitted code series $w_t = (w_{1,t}, w_{2,t}, \ldots, w_{n,t})$, the noise series $z_t = (z_{1,t}, z_{2,t}, \ldots, z_{n,t})$ is expressed by

$$z_{i,t} = y_{i,t} - w_{i,t},$$

for all $i$. Therefore, the metric for AWGN is expressed by

$$\ln P(Y|W) = \sum_{t=0}^{N-1} \ln P(y_t|w_t) = \sum_{t=0}^{N-1} \sum_{i=1}^{n} \ln P_{g2}(y_{i,t} - w_{i,t}). \quad (4)$$

The $(i, t)$-th term in Eq. (4) is represented by

$$\ln P_{g2}(z_{i,t}) = \ln \frac{1}{\sqrt{2\pi}\sigma} \exp \left( -\frac{z_{i,t}^2}{2\sigma^2} \right) = \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{z_{i,t}^2}{2\sigma^2} = \ln \frac{1}{\sqrt{2\pi}\sigma} - \frac{y_{i,t}^2 + w_{i,t}^2}{2\sigma^2} + \frac{y_{i,t}w_{i,t}}{\sigma^2}. \quad (5)$$

Let $E_b$ denote the bit energy. Then, $w_{i,t}$ is equal to $\sqrt{E_b}$ or $-\sqrt{E_b}$. Since the first, second terms and $\sigma^2$ are constant in Eq. (5), we obtain the branch metric

$$\sum_{i=1}^{n} y_{i,t}w_{i,t} = y_t \cdot w_t.$$

Therefore, the ML decoding for AWGN channel can be established to select the code series $W$ which has the maximum correlation distance to the received series $Y$. We call the Viterbi decoder optimized in AWGN channel a Gaussian Viterbi decoder.

Figure 2 illustrates the BER performances of the Gaussian Viterbi decoders over AWAN channel ($A = 0.01$ and $\Gamma = 0.01$). Figure 2 shows that impulsive noises less influence the Gaussian Viterbi decoder with hard decision than that with soft decision. This is because the hard decision gives the effect of a limiter of the impulsive noises.

3. Viterbi Decoding Optimized for Class A Noise

In this section, we propose a Viterbi decoder optimized in AWAN channel. We call our proposed decoder a Class A Viterbi decoder. In the simplified Class A noise model shown in $P_{A,\Gamma,\sigma^2}(z)$, the branch metric is approximate to

$$\ln P(y_t|w_t) \approx \sum_{i=1}^{n} \ln \frac{E_{b}}{\sqrt{E_{b}}} \cdot (y_{i,t} - w_{i,t}).$$

We deduce the Class A Viterbi decoder by using the Gaussian Viterbi decoder. As $w_{i,t}$ is $\sqrt{E_b}$ or $-\sqrt{E_b}$, the differ-
Eq. (7) is constant times Eq. (6). Hence, the Gaussian Viterbi decoder regards this difference as an input of the Gaussian Viterbi decoding. Therefore, the dotted line part in Figure 3 is represented as an input of the Gaussian Viterbi decoder. Figure 3 illustrates the concept of our proposed Class A Viterbi decoder.

We regard the difference as an input of the Gaussian Viterbi decoder. Then, the branch metric of the Gaussian Viterbi decoding is

\[
\sum_{i=1}^{n} \ln \frac{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} - \sqrt{E_b})}{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} + \sqrt{E_b})} w_{i,t}.
\]

The difference of the branch metric of a term is

\[
\ln \frac{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} - \sqrt{E_b})}{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} + \sqrt{E_b})} = \frac{\ln \frac{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} - \sqrt{E_b})}{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} + \sqrt{E_b})}}{\sqrt{E_b}} - \ln \frac{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} - \sqrt{E_b})}{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} + \sqrt{E_b})} (-\sqrt{E_b}) = 2\sqrt{E_b} \ln \frac{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} - \sqrt{E_b})}{\hat{P}_{A,\Gamma,\sigma^2}(y_{i,t} + \sqrt{E_b})}.
\]

Eq. (7) is constant times Eq. (6). Hence, the Gaussian Viterbi decoder whose input is Eq. (6) plays the same role in the Class A Viterbi decoder. Figure 3 illustrates the concept of our proposed Class A Viterbi decoder.

We analyze the dotted line part of Figure 3 in detail to realize the Class A Viterbi decoder. At first, we simplify Eq. (3) to

\[
\hat{P}_{A,\Gamma,\sigma^2}(z) = \sqrt{2\pi e^A} \hat{P}_{A,\Gamma,\sigma^2}(z)
\]

\[
= \begin{cases} 
1 \exp^{-z^2/2\sigma_0^2} = \hat{P}_b(z), & 0 \leq |z| < a, \\
\frac{A}{\sigma_1} \exp^{-z^2/2\sigma_1^2} = \hat{P}_1(z), & a \leq |z| < b, \\
\frac{A^2}{2\sigma_2} \exp^{-z^2/2\sigma_2^2} = \hat{P}_2(z), & b \leq |z|,
\end{cases}
\]

Hence, the branch metric is expressed by

\[
\ln \hat{P}_b(z) = \frac{z^2}{2\sigma_0^2} + L_0, \quad 0 \leq |z| < a,
\]

\[
\ln \hat{P}_1(z) = \frac{z^2}{2\sigma_1^2} + L_1, \quad a \leq |z| < b,
\]

\[
\ln \hat{P}_2(z) = \frac{z^2}{2\sigma_2^2} + L_2, \quad b \leq |z|,
\]

with

\[
L_0 = \ln \frac{1}{\sigma_0}, \quad L_1 = \ln \frac{A}{\sigma_1}, \quad L_2 = \ln \frac{A^2}{\sigma_2}.
\]

Since logarithmic function is a monotone increasing function, Eq. (8) is expressed by

\[
\ln \hat{P}_{A,\Gamma,\sigma^2}(z) = \max_{m=0,1,2} \left[ \ln \hat{P}_m(z) \right].
\]

Therefore, the dotted line part in Figure 3 is represented as Figure 4. We call the block diagram shown in Figure 4 a Class A preprocessing.

The Class A Viterbi decoder consists of the Class A preprocessing and the Gaussian Viterbi decoder. There is no need to configure the Gaussian Viterbi decoder. The Class A Viterbi decoder can be easily established by putting the Class A preprocessing on the front-end of the Gaussian Viterbi decoder.

4. Simulation Results

In this section, we discuss the performance of our proposed Class A Viterbi decoder through the simulation results. We design the simulation model as a low pass equivalent model. Figure 5 illustrates the block diagram of our simulation system. We compare the performance of our proposed Class A Viterbi decoder to that of the Gaussian
Figure 5: The block diagram of the simulation system.

Viterbi decoder with soft decision over AWAN channel with the impulsive index $A = 0.01$ and the GIR $\Gamma = 0.01$. The convolutional encoder is defined by Figure 1 and the traceback depth of both the Viterbi decoders is 24. Figure 6 illustrates the BER performances of the Class A Viterbi decoder and the Gaussian Viterbi decoder over AWAN channel. As shown in Figure 6, our proposed Class A Viterbi decoder has about 30dB decoding gain at $BER = 10^{-5}$ as compared to the Gaussian Viterbi decoder. Therefore, these simulation results show that our proposed Class A Viterbi decoder is useful for the Class A noise environment.

5. Conclusion

In this paper, we have discussed the Viterbi decoder for convolutional code over AWAN channel. Our proposed Viterbi decoder consists of a nonlinear preprocessing device and the conventional Viterbi decoder, and then can be easily established by putting the preprocessing device on the front-end of the conventional Viterbi decoder. Furthermore, we have compared the BER performance of our proposed Viterbi decoder with that of the conventional Viterbi decoder by computer simulation. Consequently, we have shown that our proposed Viterbi decoder has a better performance than the conventional Viterbi decoder over AWAN channel.

References


