International Journal of Computing and Business Research
ISSN (Online) : 2229-6166
Volume 2 Issue 3 September 2011
http://www.researchmanuscripts.com

MITIGATION OF INTERSYMBOL INTERFERENCE WITH EQUALIZATION AND CONVOLUTION CODING

T. L. Singal
Professor
Department of Electronics & Communication Engineering,
Chitkara University, Chandigarh, India

Abstract: In mobile radio communications, various phenomena such as multipath wave propagation, time dispersion, and fading introduce errors like Intersymbol interference (ISI), and other distortions into the signals that are transmitted over wireless channel in mobile radio communications. To combat errors, equalization (a sophisticated set of signal processing techniques), convolution coding and other schemes are employed. Equalization process compensates for channel-induced interference whereas convolution coding enable for detection and possible correction of errors in a digital mobile radio communication system. Technical computing is very powerful for simulation and evaluation of error-performance of digital data transmission. This paper gives a brief overview of adaptive equalization and the convolution coding used in mobile communication systems. Suitable algorithms for simulation of these techniques are presented here. The results are obtained using MATLAB functions and derived in the form of scatter plot and constellation of equalized signal along with filtered and ideal signal for QPSK modulation. The results show significant improvement in error-performance of digital data transmission. The analysis can be extended to include different digital modulation schemes and wireless channel characteristics.

Keywords: Communication; Convolution coding; Equalization; ISI; Simulation.

1. Introduction

The mobile radio channel puts fundamental limits on the performance of communication systems due to noise, interference, fading, and other distortions into the signals that are transmitted. In mobile communication system, a signal experiences multipath propagation which causes signal fluctuations in time. Time-dispersive wireless channels can cause intersymbol interference (ISI), that is, in a multipath scattering environment; the receiver sees delayed versions of a symbol transmission, which can interfere with other symbol transmissions. [1]. An equalizer attempts to mitigate ISI and thus improve the system's performance. The equalization process is a sophisticated signal processing technique which helps the demodulator to recover a baseband pulse with the best possible SNR, free of any ISI. [2]. Error control coding techniques detect and possibly correct errors that occur when messages are transmitted in a wireless communication system using digital transmission technology. To accomplish this, the encoder transmits not only the information symbols, but also one or more redundant symbols. The decoder at the receiver end uses the redundant symbols to detect and possibly correct the received information data whatever errors occurred during transmission. [3].

2. Adaptive equalization

There are distinct classes of adaptive equalizers such as linear adaptive equalizers and decision-feedback equalizers, each having a different overall configuration. They can use any one of the adaptive algorithms such as Least mean square (LMS), Signed LMS, Normalized LMS, Variable-step-size LMS, and Recursive least squares (RLS). [4].

A decision-feedback adaptive equalizer is a nonlinear equalizer that contains a forward filter and a feedback filter. The forward filter is similar to the linear equalizer; whereas the feedback filter contains a tapped delay line whose inputs are the decisions made on the equalized signal. Figure 1 depicts a model of a decision-feedback equalizer with L forward weights and (N-L) feedback weights.

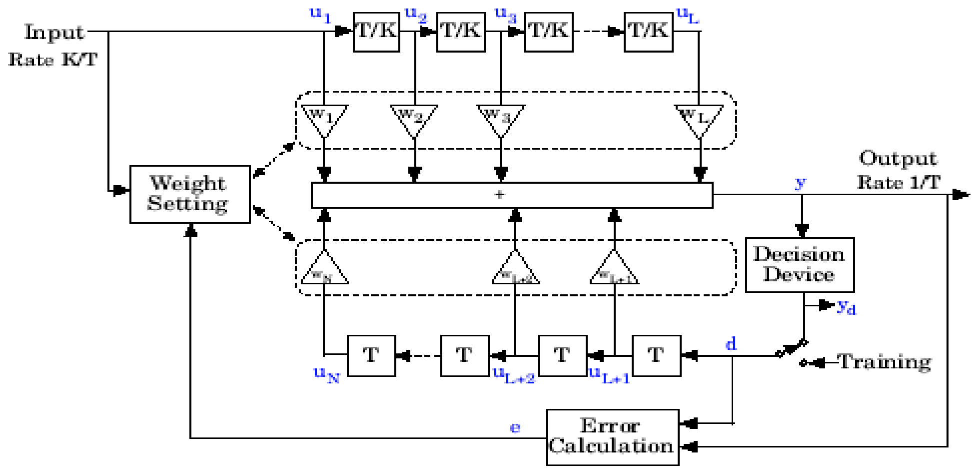


Figure 1. Model of decision-feedback equalizer

In each symbol period, the equalizer receives K input samples at the forward filter, as well as one decision or training sample at the feedback filter. The equalizer then outputs a weighted sum of the values in the forward and feedback delay lines, and updates the weights to prepare for the next symbol period.

3. Simulation using adaptive equalization functions and objects

The algorithms for the weights setting and error calculation blocks are determined by the adaptive algorithm chosen. The new set of weights depends on the current set of weights, the input signal, the output signal, and a reference signal, whose characteristics depend on the operation mode of the equalizer. The algorithm for the weight setting block in decision feedback equalizer jointly optimizes the forward and feedback weights. In typical applications, an equalizer begins by using a known sequence of transmitted symbols when adapting the equalizer weights. This training sequence enables the equalizer to gather information about the channel characteristics. At the end of processing of the training sequence, the equalizer adapts the weights in decision-directed mode using a detected version of the output signal. To use training sequence, the symbols of the training sequence is used as an input vector in simulation.

Configuring an equalizer involves choosing an adaptive algorithm and indicating the choice when creating an equalizer object in MATLAB. It contains information about the equalizer, such as the name of the equalizer class, the name of the adaptive algorithm, and the values of the weights. The equalizer object is applied to the signal that is required to be equalized, using the equalize function. At first, the reference tap index in the equalizer has a default value, but assigning a new value to the property changes this index.

The choice of adaptive algorithm depends on the specific requirements. For example, the LMS algorithm executes quickly but converges slowly, and its complexity grows linearly with the number of weights. The RLS algorithm converges quickly, but its complexity grows significantly and is suitable only when the number of weights is large. The normalized LMS and variable-step-size LMS algorithms are more robust to variability of the input signal's statistics such as power. The signed LMS algorithms simplify hardware implementation. The adaptive algorithm functions provide a way to indicate the choice of adaptive algorithm, and also allow specifying certain properties of the algorithm. A typical scatter plot as an outcome of related M-code [5] is given in Figure 2.

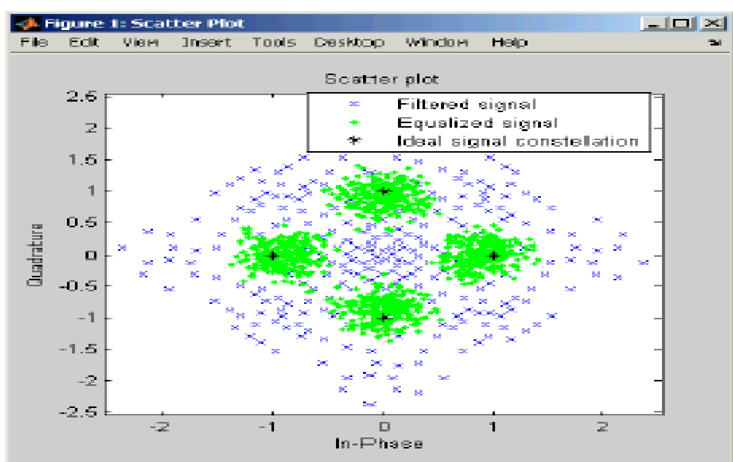


Figure 2. Scatter plot and constellation of filtered, equalized and ideal signal

The M-code produces one scatter plot for each iteration, indicating the iteration number and the adaptive algorithm, as shown in Figure 3. It may be noted from the plot that the points of the equalized signal are clustered more closely around the points of the signal constellation.

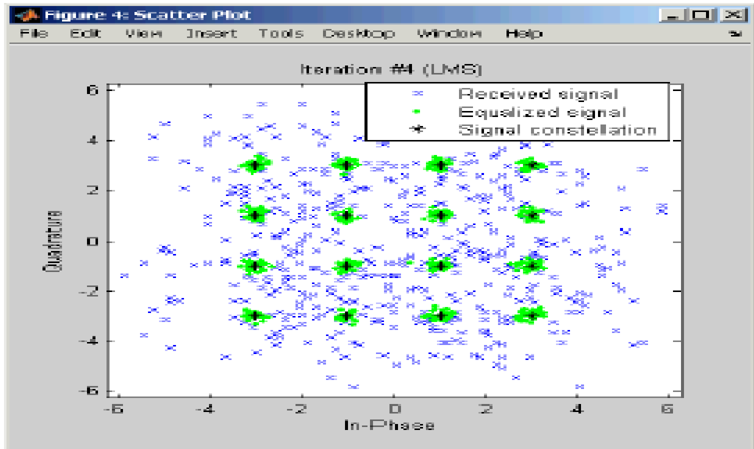


Figure 3. Scatter plot of LMS equalizer

4. Convolution coding

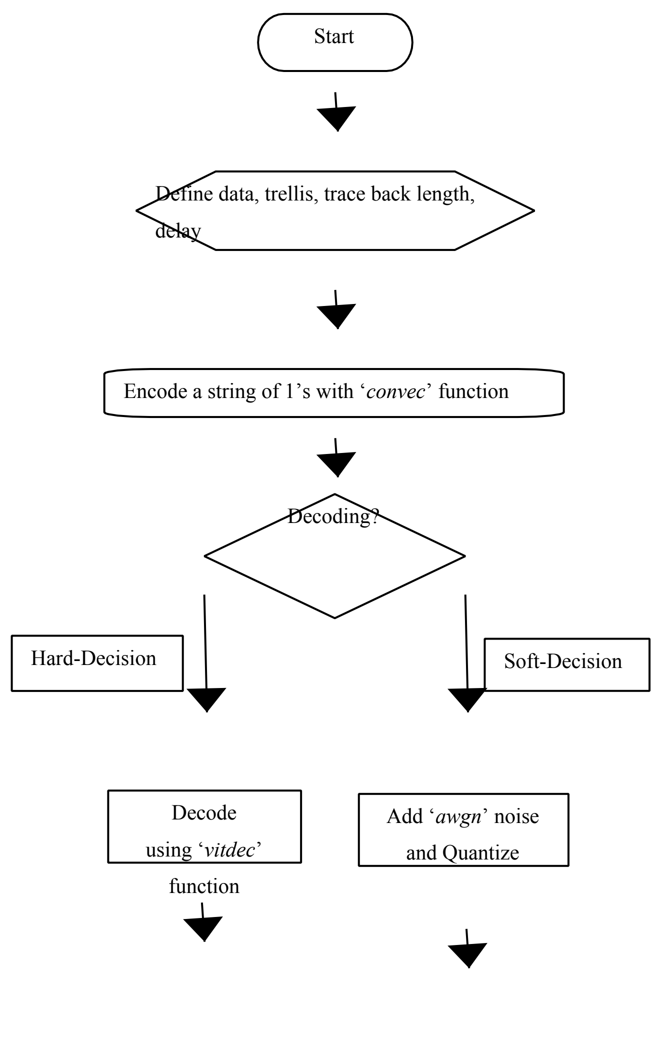
Error-control coding deals with the techniques used to enhance digital signals so that they are less vulnerable to channel impairments such as noise, interference, fading and jamming. [6]. For a given data rate, error-control coding techniques can either reduce the probability of error, or reduce the required SNR to achieve a desired probability of error P_e at the expense of transmission bandwidth or decoder complexity. Even though a convolution coder accepts a fixed number of message symbols and produces a fixed number of code symbols, its computations depend not only on the current set of input symbols but also on some of the previous input symbols. A wireless channel that exhibits multipath fading is an example of a channel with memory. [7].

A polynomial description of a convolution encoder describes the connections among shift registers and modulo-2 adders. It has either two or three components, depending on whether the encoder is a feedforward or feedback type. The components are constraint lengths, generator polynomials, and feedback connection polynomials (for feedback encoders only).

The constraint lengths of the encoder form a vector whose length is the number of inputs in the encoder diagram. The elements of this vector indicate the number of bits stored in each shift register, including the current input bits. To represent a feedback encoder, a vector of feedback connection polynomials is needed. If the encoder has a feedback configuration and is also systematic, then the code generator and feedback connection parameters corresponding to the systematic bits must have the same values [8]. If the encoder has k inputs and n outputs, then the code generator matrix is a k-by-n matrix.

5. Algorithm for creating and decoding convolution codes

The error-performance of digital data transmission using convolution coding can be simulated using MATLAB trellis structure. A trellis structure must have five fields, namely NumInputSymbols, NumOutputsymbols, NumStates, NextStates, and Outputs. It can be created using MATLAB functions such as structurename.fieldname notation, the poly2trellis function, or a single struct command. The functions for encoding and decoding convolutional codes are convenc and vitdec. Figure 4 gives an algorithm for creating and decoding convolutional codes.



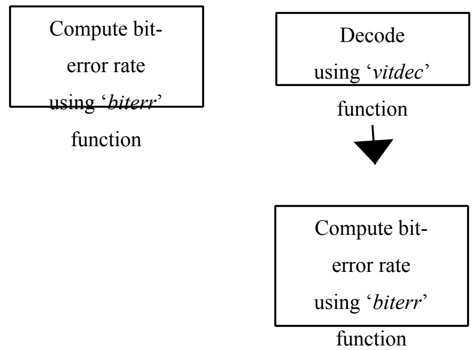


Figure 4. An algorithm for creating and decoding convolutional codes

The algorithm for creating and decoding convolutional codes illustrates decoding with hard decision or 3-bit soft decision. First it creates a convolutional code with *convenc* function and adds white Gaussian noise to the code with *awgn* function. Then, to prepare for soft-decision decoding, the algorithm uses *quantiz* function to map the noisy data values to appropriate decision-value integers between 0 and 7. Finally, the algorithm decodes the code and computes the bit error rate. Note that when comparing the decoded data with the original message, the algorithm takes the decoding delay into account. The continuous operation mode of *vitdec* function causes a delay equal to the trace back length.

To decode using hard decisions, the *vitdec* function is used with the flag 'hard' and with binary input data. Because the output of *convenc* function is binary, hard-decision decoding can use the output of *convenc* function directly, without additional processing. To decode using soft decisions, the *vitdec* function is used with the flag 'soft'. The number, *nsdec* function, of soft-decision bits must be specified and input data consisting of integers between 0 and (2nsdec-1) only should be used. An input of 0 represents the most confidant 0, while an input of (2nsdec-1) represents the most confidant 1. Other values represent less confident decisions. The output of corresponding M-code using this algorithm for creating and decoding convolutional code is [5]:

number of errors = 5

bit-error rate = 0.0013

6. Conclusions

The results for error performance in terms of the number of errors occur in trying to recover the modulated message with and without the equalizer are analyzed on simulated digital data using specified algorithm. It is observed that the linear adaptive equalizers improve the symbol error rates performance significantly. Based on simulation needs, an equalizer object can be configured in order to customize the process by varying parameters. The algorithms for creating and decoding convolutional codes are presented here using well-defined functions in communications toolbox of a technical computing language MATLAB. The corresponding M-codes show the desired results in terms of significant improvement in error performance of digital data transmission. Similar approach of developing algorithms can be extended to include source coding, digital modulation technique, interleaving, and filtering to analyze the bit-error-rate performance of digital mobile radio communication system.

References

[1] Simon Marvin K., Alouini Mohamed-Slim (2004) “Digital Communication over Fading Channels”, 2nd Edition, Wiley-IEEE Press.

[2] Poor H. Vincent, Tong Lang (2002) “Signal Processing for Wireless Communication Systems”, Springer.

[3] MacEliece Robert J. (2002) “Theory of Information and Coding”, Cambridge University Press.

[4] Singal T. L. (2010) “Wireless Communications”, 1st Edition, Tata McGraw-Hill Education.

[5] MATLAB Version 7.0.1, The MathWorks. Inc., 3, Aple Hill Drive, Natick, MA 01760-2098.

[6] Kondoz A. M. (2005) “Digital Speech: Coding for Low Bit Rate Communication Systems”, 2nd Edition [E-Book], Wiley.

[7] Soyiandah KMS, Jahmeerbacus MI (2000) “A modular approach to the design and simulation of digital communication systems,” International Journal of Electrical Engineering Education.

[8] Harada Hiroshi, Prasad R. (2002) “Simulation and Software Radio for Mobile Communications”, Artech House.

Author Profile

Prof. T. L. Singal received his B. Tech. (ECE) degree from National Institute of Technology Kurukshetra, India in 1981 and M. Tech. (ECE) degree from Punjab Technical University, India. He is the author of a text-book titled `Wireless Communications', published in 2010 by Tata McGraw-Hill Education. He has total experience of 30 years including 22 years industrial experience in the field of telecom and wireless communications in India and USA. He has convened international conference on wireless networks & embedded systems in 2009. He has coordinated several technical workshop courses under faculty leadership institute programs of Indo US Collaboration of Engineering Education (IUCEE) during 2010-2011. His major areas of interests are Analog & Digital Communications, Cellular and Mobile Communications, Wireless Communication Systems & Networks.