DESIGN OF ARQ AND HYBRID ARQ PROTOCOLS FOR WIRELESS CHANNELS USING BCH CODES

Kalpana Chikatwar¹, Ramesh D², Satish kannale³

¹, ², ³ Department of Electronics and Communication Engineering
Bhimanna Khandre Institute of Technology, Bhalki-585328

ABSTRACT

Automatic retransmission request (ARQ) is a feedback-based data link layer technique which enhances the reliability of communication in fading channels. In this paper, we have discussed the performance features of ARQ protocols for wireless channels and also compared their performance in terms of throughput efficiency. Further we have designed and implemented a hybrid ARQ scheme for wireless channels for improving the performance of ARQ scheme using FEC scheme (where BCH code is used as forward error correcting code). The designed hybrid-ARQ provide increase in the system throughput efficiency and reliability of the system.

Keywords: ARQ, Wireless channels, Forward Error Control (FEC), Hybrid ARQ

I. INTRODUCTION

As we know that there are two categories of techniques for controlling the transmission errors in the data transmission systems. The forward error control (FEC) scheme. Automatic-repeat-request (ARQ) scheme.

In an FEC system, an error correcting code is used. When receiver detects the presence of error in a received vector it attempts to determine error location and then corrects the error . If exact locations of the errors are determined, the received vector will be correctly decoded if receiver fails to determine the exact locations of the errors, the received vector will be decoded incorrectly, and erroneous data will be delivered to the user. Hence it is hard to achieve high system reliability with FEC [3] (because probability of decoding error is much greater than prob. of undetected error).

In an ARQ system a code with good error detecting capability is used. At the receiver syndrome of the received vector is computed. If the syndrome is zero, the received vector is assumed to be error free and is accepted by the receiver. At the same time receiver notifies the transmitter via return channel that the transmitted codeword has been successfully received. If the syndrome is not zero, errors are detected in the received vector. Then the transmitter is instructed, through the return channel to retransmit the same codeword . This retransmission continues until the codeword is
successfully received [6]. With this system erroneous data are delivered to the data sink only if receiver fails to detect the presence of errors. If proper linear code is used the probability of an undetected error can be made very small. Hence ARQ is simple and provide high system reliability.

II. HYBRID ARQ SCHEME

The major advantage of ARQ over FEC is that error detection requires simpler decoding algorithms then the error correction. Hence ARQ is simple and provide high system reliability. However the throughput of ARQ systems falls rapidly with increase in channel error rate [1].

System using FEC maintains constant throughput (equal to the code rate R = k/n) regardless of channel error rate. But it is difficult to achieve high system reliability using FEC, because error correction requires complex decoding algorithms than the error detection. Hence this makes decoding hard to implement and expensive [3]. The drawbacks in both ARQ and FEC could be overcome if two error control schemes are properly combined. Such a combination of two basic error control schemes is referred to as hybrid ARQ. A hybrid ARQ system consists of an FEC subsystem contained in an ARQ system.

The function of the FEC subsystem is to reduce the frequency of retransmission by correcting the error patterns that occurs most frequently. This increases the system throughput. When a less frequent error pattern occurs and is detected, the receiver requests for the retransmission rather than passing the unreliably decoded message to the user. This increases the system reliability. As a result, a proper combination of FEC and ARQ provides higher reliability than the FEC system alone and a higher throughput than the system with ARQ only. Furthermore, since the decoder is designed to correct a small collection of error patterns, it can be simple.

III. TYPE-1 HYBRID ARQ PROTOCOL

A straightforward hybrid ARQ scheme is to use a code, say an (n, k) linear code, that is designed to simultaneously detect and correct errors. When a received vector is detected in error, the receiver first attempt to locate and correct the errors. If the number of errors (or the length of the error burst) is within the designed error correcting capability of the code, the errors will be corrected and the decoded message will be passed to the user or stored in buffer until it is ready to be delivered. If an uncorrectable error pattern is detected, the receiver rejects the received vector and request for the retransmission. When the retransmitted vector is received the receiver again attempts to correct the errors. If the decoding is not successful the, the receiver again rejects the received vector and asks for another retransmission. This error correction and retransmission process continues until the vector is successfully received or decoded. This scheme is best suited for channels where the level of noise and interference is fairly constant.

For example, we can use the (1023, 923) BCH code in the hybrid ARQ system. This code has a very good error detecting and error correcting capability. This code has a minimum distance of \(d_{\text{min}}=21\) it can be used for correcting five or fewer errors. If an error pattern with five or fewer errors occurs it will be detected and corrected. If an error pattern with more than 5 but less than 16 errors occurs, it will be detected and in this event receiver will request for the retransmission of the erroneous vector.

IV. BCH CODE DESIGN

To construct a t-error correcting q-ary BCH code of length n:

1. Find a primitive \(n^{th}\) root of unity \(\alpha\) in a GF \((q^m)\) where m is minimal.
2. Select \((\delta-1) = 2t\) consecutive powers of \(\alpha\), starting with \(\alpha^b\) for some non-negative integer \(b\).
3. Let \(g(x)\) is the least common multiple of the minimal polynomials for the selected powers of \(\alpha\) with respect to GF \((q)\).
Thus generator polynomial \( g(x) \) of a \( t \)-error-correcting primitive BCH codes of length \( 2^m - 1 \) is given by

\[
g(x) = \text{LCM} \{ \varphi_1(x), \varphi_2(x), \ldots, \varphi_{2^t}(x) \}
\]

Where \( \varphi_1, \varphi_2, \ldots \) are the minimal polynomials of \( g(x) \)

**Step 1** follows from design procedure for general cyclic codes. **Step 2** and 3 ensures, through the BCH bound, that the minimum distance of the resulting code equals or exceeds \( \delta \) and the generator polynomial has the minimal possible degree.

**Systematic encoding algorithm for an \( (n, k) \) BCH code:**

As we know that BCH codes are basically cyclic codes hence they are encoded using the same procedure as that of used for encoding cyclic codes. In summary, encoding in systematic form consists of three steps [3]

**Step 1:** Multiply the message polynomial \( m(x) \) by \( x^{n-k} \)

**Step 2:** Divide the result of step 1 by generator polynomial \( g(x) \). Let \( d(x) \) is the remainder obtained from this division.

**Step 3:** Encoded codeword \( c(x) = m(x).x^{n-k} - d(x) \).

**DECODING OF BCH CODES:**

**Peterson’s direct solution decoding algorithm for \( t \)-error correcting BCH code:**

**Step 1:** Compute the syndromes \( S = (s_1, s_2, s_3, \ldots, s_{2t}) \) from the received polynomial \( r(x) \).

\[
S = r(\alpha^i) \text{ for } 1 \leq i \leq 2t
\]

**Step 2:** set \( v = t \) and construct the syndrome matrix \( M \).

\[
M = \begin{bmatrix}
S_1 & \cdots & S_v \\
\vdots & \ddots & \vdots \\
S_v & \cdots & S_{2v-1}
\end{bmatrix}
\]

**Step 3:** compute the determinant of the syndrome matrix \( M \). if \( \det(M) \approx 0 \) go to step 5.

**Step 4:** If \( \det(M) = 0 \) (singular matrix) set \( v = t - 1 \) and construct the new \( M \) matrix again and check for nonzero determinant. This procedure is repeated until we get \( \det(M) \approx 0 \).

**Step 5:** Find the inverse of \( M \) matrix and solve the following equation for \( \Lambda \) and construct \( \Lambda(x) \)

\[
\begin{bmatrix}
\Lambda_v \\
\Lambda_{v-1} \\
\vdots \\
\Lambda_1
\end{bmatrix} = M^{-1} \begin{bmatrix}
-S_{v+1} \\
-S_{v+2} \\
\vdots \\
-S_{2v}
\end{bmatrix}
\]

**Step 6:** Find error location by finding roots of the \( \Lambda(x) \) (using Chine’s search procedure).

**Step 7:** Inverse of the roots of the \( \Lambda(x) \) gives the exact error location number.

**Step 8:** Subtract the error polynomial \( e(x) \) from the received polynomial \( r(x) \) to obtain exact decoded codeword.

Decoded codeword = \( r(x) - e(x) \)
V. SIMULATION RESULTS

This section discusses the performance factors and simulation results of different ARQ techniques (including both basic ARQ and hybrid ARQ schemes). These performance factors help to study how ARQ schemes can be used for the different applications. The following performance factors are considered.

1. Throughput efficiency ($\eta$)
2. Signal to Noise Ratio (SNR)

1. Throughput efficiency ($\eta$)
   
   Throughput efficiency is the measure of performance of an ARQ system which is defined as the ratio of average number of information code words successfully accepted by the receiver per unit of time to the total number of code words that could be transmitted per unit of time. All three basic ARQ schemes achieve same reliability; however they have different throughput efficiencies[1]. From the results obtained we can draw following conclusions
   
   1. Throughput performance of Stop-and-wait ARQ protocol is least than the continuous ARQ protocol because of the ideal time spent in waiting for the acknowledgment for each transmitted codeword. Stop-and-wait ARQ is basically designed for use on half duplex channels.
   
   2. Throughput performance of Go-back-N ARQ protocol is greater than stop-and-wait ARQ protocol. This is mainly due to the continuous transmission and retransmission of code words from the transmitter. But it is less than the selective repeat ARQ protocol because of the retransmission of the many error free code words following the codeword detected In the error Go-back-N ARQ is basically designed for use on full duplex channels.
   
   3. Throughput performance of Selective repeat ARQ protocol is higher among the all ARQ schemes. This is because; here only the code words which are detected in error are retransmitted and throughput of Selective repeat ARQ does not depend on the round trip delay.[7]
   
   4. Throughput performance of hybrid ARQ protocol is good at higher values of SNR and at lower values of SNR the throughput is nearly zero. (From 1db to 7.5db the throughput is =0) This is because the errors introduced by the channel are more then the error correcting capability of the code. Hence we can conclude that hybrid ARQ protocol provide higher throughput performance and higher system reliability than the ARQ system alone.

2. Signal to Noise Ratio (SNR)

   The most direct factor that affects the performance of the ARQ technique is the signal to noise ratio. For a fixed received signal power, noise level can be represented by the signal energy per bit to the noise power spectral density ($E_b/No$). As ($E_b/No$) increases, the noise level decreases. Thus, the performance of the ARQ technique improves or the bit error rate (BER) decreases as ($E_b/No$) increases [5]. To analyse how the noise level affects the performance of the ARQ technique, a number of simulation tests were conducted and results are analysed.
Simulation results of basic ARQ schemes

![Comparison of all three basic ARQ schemes.](image)

Fig: Comparison of all three basic ARQ schemes.

Simulation results of Hybrid ARQ scheme using BCH code

![Simulation results obtained for hybrid ARQ scheme using (1023, 923) BCH code with N=100.](image)

Fig: Simulation results obtained for hybrid ARQ scheme using (1023, 923) BCH code with N=100.

VI. CONCLUSION

In many real time applications, ARQ techniques are often preferred to FEC schemes since they offer a simpler implementation (because error detection requires simpler decoding algorithms than the error correction) and a higher performance for moderate and high signal-to-noise ratios. At the same time, ARQ techniques perform well on many types of real channels for e.g. packet switching data n/w, computer communication n/w and adapt in a satisfactory way to different behaviours on the transmission channel.

The major advantage of ARQ over FEC is error detection require simpler decoding algorithms then the error correction hence ARQ is simple and provide high system reliability. However their throughput falls rapidly with increase in channel error rate. System using FEC maintains constant throughput regardless of channel error rate. But it is difficult to achieve high system reliability using FEC because error correction require complex decoding algorithms than the error detection hence this makes decoding hard to implement and expensive.
For improving the performance an ARQ scheme using FEC scheme (where BCH code is used as forward error correcting code) is designed. Such a combination of two basic error control schemes is called as HYBRID ARQ SCHEME which provide increase in the system throughput efficiency and reliability of the system. Hence from the obtained simulation results we can conclude that hybrid ARQ protocol provide higher throughput performance then the simple ARQ scheme alone

VII. REFERENCES