Trevor de Clercq
October 27th, 1998

Digital Audio Processing I
E85.2600, Prof. K. Peacock

Chapter 6 "Art of Digital Audio" Abstract

Despite John Watkison's assurance in the beginning of Chapter 6 that a non-mathematical description of digital audio error correction would be "quite straightforward," I wasn't too surprised to find this topic, similar to others presented in his text, quite a prickly subject. Errors are a natural byproduct of any recording system, analog or digital, represented in the medium as noise. There are two basic types of erros which plague digital recordings, burst errors and random errors. Burst errors affect a large group of bits and are usually the result of dirt or decomposition of the medium; random errors, on the other hand, are defined as only affecting single bits or symbols. It is the fundamental advantage of digital recording that errors introduced into the recording system can be identified and corrected, as compared to the irrevocable subtle degradations inherent to analog.

The basic mechanism of error correction operates upon the principle of redundancy. Extra bits are added to recorded information as a means of checking against the addition of errors. These extra bits are calculated from the original information itself, and, when combined with the data itself, are termed codewords. Theoretically speaking, the longer the codeword (and the more added redundancy), the higher the chance of detecting and correcting errors becomes. Since digital audio is by nature a real-time operation, however, higher rates of information transmission are more difficult to implement. The art of error correction, therefore, is to create an efficient method of data encoding which balances redundancy with transmission speed.

In order to correct errors, the errors themselves must first be identified. In fact, since digital audio relies on the binary number system, error correction itself is a trivial matter. To correct a bit in error, the state of the bit is merely reversed; thus a one becomes a zero and a zero becomes a one. Error correction, therefore, should be more aptly termed error detection. The method of error detection relies on the addition of parity bits to the original data. A simple example of the calculation of a parity bit would be an XOR gate. The result from this XOR gate is added to the data and thus forms the codeword.

The coding mechansims for error detection, however, are much more complicated than simple XOR gates. The foundations for practical parity calcuation are known as block coding and convolutional coding. Block coding arranges the data into a two-dimensional array. Parity bits are generated for both the rows and the columns of this array. Error correction by the block method is therefore identifiable at the intersection of two failing checks. Convolutional coding, on the other hand, relies on previous data to calculate the current parity bit. Through this method, a continuous stream of interlocking data and parity bits are created on an endless basis. Because of the boundless nature off convolutional coding, editing becomes quite difficult. Block coding, as a result, is favored in recording situations whereas convolutional codes are reserved for other applications, such as digital audio broadcasting.

In order to effectively judge the error detecting capabilities of a system, the understanding of Hamming distance becomes useful. The Hamming distance of a coding system is defined as the minimum number of incorrect bits which would change one codeword into another. It is essential to guard against codewords shifting from one to another so that the error correction system does not start miscorrecting errors or changing correct bits into errors. Distance 3 Hamming codes, for example, need three errors to change one codeword into another. A codeword with a single-bit error will thus be correctly identified. A codeword with a double-bit error, however, will be incorrectly identified as a different codeword and therefore be miscorrected to match this other codeword. In general, Hamming distance relates the point after which the coding system breaks down and begins corrupting data.

Probably one of the most effective means of avoiding errors is through interleaving of the codewords. With interleaving, data is dispersed throughout a variety of tracks on the recording medium. A simple example of interleaving would be allocating every other sample to an odd track and all the others to an even track. With this example, if one of the tracks is wiped out, the data can still theoretically be reconstructed since every other sample still exists on the intact track. The original waveform could thereby be reproduced through basic concealment measures. In practical usage, interleaving methods go far beyond this simple example in order to minimize the effect of dropouts and other burst errors which can potentially corrupt larger portions of information.

A high level of error correction and a complicated use of interleaving combine to form the Cross-Interleave Reed-Solomon Code (CIRC). CIRC is of particular interest to me since it is the most commonly used encoding method of RDAT samples. A full understanding of the complete mathematical underpinnings of Reed-Solomon coding is beyond my current research capabilities; however, I was able to gather from Kenneth Pohlmann that RS codes are based on Galois fields. With Galois fields, a matrix is used in which each element can be the sum and/or product of any other element. This formula for error correction is strengthened by cross-interleaving, which is basically double interleaving. First, the input samples are parity encoded and interleaved. Second, the interleaved samples are RS coded and then interleaved again. The effect of cross-interleaving is to break up random errors as well as burst errors, thus providing a more impenetrable system of error detection.

As I stated before, the study of error correction and detection compromises are a large body of knowledge. Hopefully with this chapter, I have at least begun in the right direction to fully understanding the coding of digital audio information and thus towards a fuller understanding of digital audio itself.