White Paper
FEC In Optical Transmission

Giacomo Losio
ProLabs Head of Technology
FEC In Optical Transmission

When we introduced the DWDM optics, we left out one important ingredient that really makes the difference: Forward Error Correction (FEC). FEC is what permits networking equipment to receive a degraded signal, and still deliver error free traffic to the end user, correcting transmission errors. The basic behaviour is fairly simple, the implementation and mathematics behind it can be definitely complicated. FEC works like when we read a text and there is a typo. In most words if the order of 2 letters is changed we have no difficulty to understand which one the original word was. If typos are rare enough, we have no problem reading the text, if things get too messed up, the message can't be understood. FEC works in a similar way. One might argue that there are simpler methods to achieve the same target (error free transmission). If the problem is that the signal is too low, the transmit power might be increased, right? False: fiber is not a linear medium and beyond a certain level other penalties start to become relevant. Reducing amplifier noise is also complicated, after all, as often happens, optics needs electronics help.

The FEC function is split between transmitting and receiving equipment: The transmitter encodes the data and adds a redundancy that will be used by the receiver to detect and correct error. Receiving block decodes the data and - based on the redundancy transmitted is able to localize errors and correct them up to a certain threshold.

In this paper we will not talk about the mathematical foundations of the FEC codes, but we need a little terminology and a few concepts to be able to understand what people really mean when they talk about this topic.

Most FEC implementations intended for optical networking use block codes. Block codes divide the incoming data stream into blocks of information bits and append parity bits to each block to create an encoded pattern of bits making the codeword. A codeword is transmitted in its entirety and is used to detect and correct errors at the receiving end. The greater the number of parity bits, the greater the correction ability of a code, but the transmission rate must also increase as more bits are packed into the same time slot. The challenge is to correct as many errors as possible while maintaining low parity overhead. FEC algorithms vary in terms of parity overhead, correction capability, and burst-error tolerance. The block encoder takes a block of k bits and replaces it with a n-bit codeword (n is bigger than k).
For a binary code, there are a possible $2^k$ codewords in the codebook. The channel introduces errors and the received word can be any one of $2^n$ n-bit words of which only $2^k$ are valid codewords. The job of the decoder is to find the codeword that is closest to the received n-bit word.

The decoding spheres of the figure below will be used to illustrate the decoding process. In Fig 1, each valid codeword is represented by a point surrounded by a sphere of radius $t$ where $t$ is the number of errors that the code can correct. Note that codewords A and B of Figure 2 are separated by a distance $d$, called the minimum distance of the code (or Hamming distance). The minimum distance of a code is defined as the smallest number of places that any two codewords in the codebook differ. Usually, codes with large minimum distance are preferred because they can detect and correct more errors.

Comparing the spheres surrounding codewords A and B in Fig 1. We see that the error correcting capability of a code is given by $d_{\text{min}} = 2t + 1$ (this is the minimum separation that prevents overlapping spheres).

Or in other words, a code with $d_{\text{min}} = 3$ can correct all patterns of 1 error, one with $d_{\text{min}} = 5$ can correct all patterns of 2 errors, and so on.

**Fig 1.** Codewords, arbitrary words and distance between codewords

**Error decoding cases**

**Case #1:** Correct decoding. Assume that codeword C is sent and the n-bit word Y is received. Because Y is inside C’s sphere, the decoder will correct all errors and error correction decoding will be successful.

**Case #2:** Decoding failure. This time we send codeword C and the channel gives us n-bit word W. The decoder has no way of knowing that codeword C was sent and must be decoded to B since W is in B’s sphere. This is an example of error correction decoder failure.

**Case #3:** Error detection without correction. This case shows one way that an error correction code can be used to also detect errors. We send codeword C and receive n-bit word X. Since X is not inside any sphere, we won’t try to correct it. We do, however, recognize that it is an errored codeword and report this information to the user.
ITU-T G.709 standard includes the definition of an FEC algorithm usually referred to as RS(255,239). This indicates the algorithm (Reed-Solomon in this case) the length of the FEC codeword (data + parity = 255 bytes) and the length of the information symbols (239 bytes), the difference between these two numbers indicates the amount of redundancy – also called overhead OH - of this scheme (~7%). In our specific case symbols are bytes. Applying the concept learned above, the RS(255,239) code can correct up to 8 symbols per codeword.

FECs have proven to be effective in extending the capabilities of optical networks that additional algorithms have been developed (the so called Enhanced FEC) that provide additional correction capability, a number of these has been described in ITU-T recommendation G.975.1. The most used techniques include concatenated codes (two cascaded codes, an inner one and an outer one) or turbo product codes that decode the incoming data stream iteratively, the errors left by iteration n might be corrected by iteration n+1.

Interleaving
One other important technique is interleaving. An interleaver is a device that takes symbols from a data stream as the input and produces the identical symbols at the output in a different temporal order. The classical use for interleaving is to disperse sequences of bits in a bitstream so as to minimize the effect of burst errors introduced in transmission.
Interleaving data from different codewords improves the efficiency of codes because the effect of burst errors is shared among many other codewords. By interleaving, it spreads the impact of a noise burst over multiple symbols, which come from different codewords. As long as each deinterleaved codeword has fewer errors than it can correct, the interleaved group of codewords will be corrected. Interleaving actually integrates the error correction powers of all of the codewords included in the interleaved group that is the depth of the interleaver.

This allows a higher rate of code and channel efficiency and still protects against an occasional very long error. For example, if 64 codewords that can correct 8 errors each are interleaved, the interleaved group can correct almost any combination of symbol errors that total less than 512. It does not matter if all 512 are in one long burst.

![Diagram](image.png)

**Fig 4.** Example of robustness against burst errors

The key metric used to describe which the “power” of an FEC is, is the coding gain, which is the difference in input signal-to-noise ratio (or “Q-factor”) for a given BER between the data with FEC encoding and the raw data.

\[
CG(dB) = 20 \log(Q_{out}) - 20 \log(Q_{in})
\]

The “net coding gain” takes into account also the bit rate increase due to the addition of redundancy.

\[
NCG(dB) = 20 \log(Q_{out}) - 20 \log(Q_{in}) - 10 \log(n/k)
\]
Fig 5. Comparison between G.975.1 4 code and standard G.709 RS(255,239)

Fig 6. G.975.1 4 algorithm coding gain
Another interesting way of seeing the effect of FEC is comparing optical eyes at levels of OSNR which are one coding gain apart.

Believe it or not, after FEC the second eye, is applied the same has BER, the first one!

The gain provided by FEC can be used to:

- Increase the maximum span length and/or the number of spans, resulting in an extended reach.
- Compensate fiber non-linearity.
- Increase the number of channels in a DWDM system which is limited by the output power of the amplifiers by decreasing the power per channel and increasing the number of channels. Decreasing channel power also helps fighting the non-linear effects.
- Relax the component specification (e.g. launched power, mask margin, extinction ratio, etc.)
- Allowing to overcome the impairments introduced by ROADM nodes, increasing the number of network elements that can be cascaded before regeneration, enabling reconfigurable optical networks.

The advancements in transceiver technology lead to the availability of DWDM optics also in pluggable form factor, performance of such devices have to be verified also in OSNR limited condition and with dispersion. “DWDM” does not mean only “precise wavelength”, a poor performance device can reduce the gain provided by FEC. Performance has to be verified also at high bit error rates typical of FEC application, extensive testing and test beds designed for DWDM application are a must.

References
