Abstract

Error-correcting codes where messages are represented by integers are termed as **"integer codes"** by Vinck and Morita (IEICE Trans., 81(20), 2013-2018, 1998). The codes are defined over the ring of integers modulo $m, \mathbb{Z}_m$. In areas where messages are represented by integers, such as coded modulation and magnetic recording, the idea of integer codes is intriguing. The use of integer arithmetic here provides a simpler mechanism in the error-correcting procedure. In integer codes, we first identify the common types of errors occurring in the channel and accordingly construct a code capable of correcting the errors. This can be considered another benefit of these codes over the traditional block codes. These codes are constructed with the help of a predetermined parity check matrix.

In the coding literature, binary-oriented codes are mainly studied. So, to deal with integer codes having codewords in binary form, Radonjic and Vujicic (IEEE Trans. Comput. 62(2), 411-415, 2013) developed a novel class of integer codes over the ring of integers modulo $2^b - 1$, $\mathbb{Z}_{2^b-1}$. They considered the ring $\mathbb{Z}_{2^b-1}$ since any non-trivial integer from it has a unique b-bit byte form written with the help of its binary representation. By doing so, the integer codes become capable of correcting errors of the form $1 \to 0$ and $0 \to 1$, which can be later converted into integers after error-correction. There are many communication channels where errors of the form $1 \to 0$ are only observed. Optical networks without optical amplifiers are an example of this. These errors are called asymmetric errors and are modelled by the Z-channel with crossover probability ϵ. However, in some LSI/VLSI ROM and RAM memories, only one type of error, either $1 \to 0$ or $0 \to 1$, is observed. These errors are called unidirectional errors and are modelled by a binary symmetric channel with crossover probability ϵ.

Errors during transmission may occur in a clustered form, called bursts. Further, other burst types of errors such as CT-bursts, low- and high-density CT-bursts, solid-bursts, etc., are studied depending upon the nature of the channel and requirements. In this thesis, we have studied these bursts according to their asymmetric, unidirectional, and symmetric nature over the ring of integers $\mathbb{Z}_{2^b-1}$. In our study, we have presented asymmetric CT-burst, low- and high-density CT-burst correcting integer codes where the errors occur within a b-bit byte. The codes are constructed with the help of computer search results and can be interleaved without the use of any additional hardware. Furthermore, we have presented unidirectional solid burst correcting integer codes where the errors occur within a b-bit byte. By extending the error-correcting capability of the integer codes for errors occurring between two b-bit bytes, we have presented two classes of integer codes capable of correcting asymmetric solid bursts and asymmetric bursts occurring anywhere in the codeword. These classes of integer codes are not required to be interleaved to correct the asymmetric bursts.

In the coding literature, codes are considered efficient in some way over other existing codes correcting similar types of errors if they are rate efficient and consume less memory during implementation. Keeping this in mind, for all of the codes presented in our study, we have tried to improve the code rate and memory consumption of the codes. These codes are compared with other similar error-correcting linear and integer codes and are found to be more efficient. For all of the codes presented, we have determined the probability of erroneous decoding and Bit Error Rate (BER), and analysed the changes with respect to the different code rates. Our study ends with the derivation of the probability of erroneous decoding and BER for a few other existing integer codes capable of correcting symmetric burst errors, single symmetric and asymmetric burst errors, sparse byte errors within a b-bit byte, and double and single asymmetric errors within and between two b-bit bytes.