Adaptive encoding in remote digital telemetry and command systems

A G Samoylov, S A Samoylov, V S Samoylov and A P Galkin

Vladimir State University named after Alexandr and Nikolay Stoletovs, Vladimir, Russia

E-mail: ags@vlsu.ru

Abstract. The article considers the priority objective of reducing data transfer time in remote digital telemetry and command networks. To achieve this objective, some adaptive encoding principles have been proposed. There are expressions allowing to determine the optimal number of redundant components for different sizes of data blocks with a given noise resistance and data transmission bit rate. The paper proposes several flow charts of decoders and encoders using remote command systems with adaptive coding.

1. Introduction

In the meanwhile, digital communication methods are widely used in various data transmission systems [1-3]. Similar methods are used both in continuous data transmission systems and in remote telemetry and control systems. The peculiarity of remote control systems is that communication sessions are rare and of short duration, which are contrasted with continuous circulation systems in trunk channels.

Digital data transmission techniques have a number of advantages, one of which is the use of noise-resistant encoding. The choice of encoding technique and its parameters depends on the requirements and operating conditions relevant to a particular communication system. If data flows have not been changed over a long period of time, then the parameters of noise-resistant codes have also remained unchanged. When the volume of transmitted data varies with each communication session, it becomes possible to decrease data transmission time, which is very important for many remote command systems [1]. This paper focuses on the ways to address the problem.

2. Materials and methods

To solve the task, it is proposed to change the parameters of the noise-proof code used, depending on the amount of data transmitted in each specific communication session. In this case, the noise-resistant encoding becomes an adaptive one [2]. The peculiarities of such encoding technique will be exemplified by Reed-Solomon (R-S) code - one of the most effective block noise-resistant codes, capable of correcting group errors as well. The basic parameters of block encoding are the block length \(n\) and the number of information symbols \(k\) [1]. Since R-S code is not a binary code, we can introduce the notion of an error symbol, not a bit. The use error probability in calculus somewhat distorts the statistical properties associated with the probability of several erroneous bit in a symbol. However, minor statistical deviations in probability of errors in a data block will hardly affect the consideration of adaptive encoding features [2].
If the data bit error probability is \( p_1 \), then the probability of an erroneous byte for an eight-bit symbol is:

\[
P_b = 1 - (1 - p_1)^8 = 8p_1 - \sum_{i=2}^{8} C_i^8 (-1)^i p_1^i \approx 8p_1,
\]

where \( C_i^8 = \frac{j!}{i!(j-i)!} \).

Most communication channels feature slow enough data bit error probability to say that \( P_b = 8p_1 \). Such a consideration does not take into account the peculiarities of possible grouping of data bit errors, but in the absence of a priori information on the statistical properties of error grouping, this simplification is entirely permissible.

The error correcting capability of a noise-resistant code is determined by the number of redundant symbols \( b = n - k \). The redundant symbols are added to the data block while generating code sequence for further transmission. As a rule, redundant symbols are found either at the beginning or at the end of the encoded data block and such an arrangement is known in advance for both the transmitting and receiving sides. R-S code allows correction of \( b/2 \) erroneous symbols, if the arrangement of symbols is not known in advance [3]. Therefore, when encoding, the value of \( b \) is chosen to be even and will be assumed as even in further calculations.

In the case where the number of erroneous symbols in a block is greater than \( b/2 \), the majority of specialized R-S codecs (for example, in ANA4011c), such a block cannot be decoded and transferred unchanged to the output. In addition, error spreading is possible when the decoder makes a wrong decoding decision. Such situations occur when the number of errors is very large and data is so distorted that the whole code word becomes completely similar to another code word outside the Hamming distance. However, such situations are so rare that they practically do not affect R-S code noise resistance calculations.

The choice of \( k \) and \( b \) parameters for R-S code is not obvious. On the one hand, increasing the redundancy \( b \) increases the noise resistance, on the other hand, the encoding bit rate \( v = k/(k + b) \) and, consequently, the transmission bit rate are decreasing [1]. Conversely, in order to achieve a constant encoding bit rate at relatively high redundancy values \( b \), it is necessary to increase the size of the data block \( k \), or the size of the code block in whole. However, increasing the size of the data block \( n \) leads to an increase in computational cost, time delays, duration of auxiliary read/write operations, etc. [3].

Adaptation of data blocks of irregular length \( k \) involves choosing such a redundancy size \( b \) so that, with a given noise resistance of the communication system, the maximum encoding bit rate is provided (i.e. the maximum transmission bit rate) [2]. To determine the optimal parameter \( b \) as a function of \( k \), consider the noise resistance of R-S code as a function of \( k \) and \( b \), regarding occurrences of erroneous symbols as independent events.

3. Results

Error probability \( N \) in a communication channel within the length interval \( n \) corresponds to the binomial distribution:

\[
P[N] = C_n^N P_b^N (1-P_b)^{n-N},
\]

where \( P_b \) is determined by the expression (1), and the average number of erroneous bytes is determined as [4]:
The average data bit error probability for a block is:

\[ P_{B,av} = \frac{N_{av,1}}{n} = P_B. \]

The probability of a successful block correction is \( P_2 \), and \( P_3 \) is the probability of a failed block correction. These probabilities can be calculated as:

\[ P_2 = \sum_{i=0}^{n-k} \binom{n}{i} P_B^i (1-P_B)^{n-i}, \quad P_3 = \sum_{i=n-k+1}^{n} \binom{n}{i} P_B^i (1-P_B)^{n-i}. \]

The average number of erroneous symbols in a block is:

\[ N_{av,2} = \sum_{N=n-k+1}^{n} NC_N^n P_B^N (1-P_B)^{n-N}. \]

In this case, the probability of a failed block correction is:

\[ P_s = \frac{1}{n} \sum_{N=n-k+1}^{n} NC_N^n P_B^N (1-P_B)^{n-N}. \] (2)

The ratio \( L = \frac{P_s}{P_B} \) is an indicator of the effectiveness of noise-resistant encoding [1]. Thus, the choice of optimal encoding parameters is based on the dependencies \( P_s(P_B) \) and equipment performance. Dependences \( P_s(P_B) \) for fixed \( k \) and various redundancy values \( b=2t \) (where \( t \) is the number of errors to be corrected) have been calculated in accordance with expression (2) and shown in figure 1a.

The above graphs confirm a significant improvement in noise resistance depending on the redundancy increase. Fig. 1b shows a family of dependencies of the symbol error probability \( P_s \) on the encoding bit rate \( V \) for different lengths of the data part \( k \) and fixed values of the bit error probability \( P_B \). The probability of a bit error in the communication channel has been assumed to be \( P_B=10^{-5} \). The length of the data block \( k \) varied in a range of 2–235 bytes.
Figure 1. Dependence of the symbol error probability for a noise-resistant R-S code on the error probability in the communication channel (figure 1a) and on the encoding bit rate (figure 1b).

The obtained analytical expressions and graphical dependencies allow adaptively to choose such a redundancy length $b$ for various data blocks, at which the given noise resistance of the communication system is reached at the maximum transmission bit rate.

4. Discussions
Consider an example of a remote command communication system design using adaptive encoding. Commands of a certain length are sent to check points (CPt) from the control panel (CPn). A CPt sends a response to CPn of a not fixed size, since the data from the CPt may be of a different length. Since CPn signals and CPt responses are of different length, CPn's and CPt's codec circuits have also to be different.

Control point encoder is a typical encoder comprising an input and output controllers, a codec and a command unit (CU). The data stream divided into fixed-length commands arrives through the controller and Z buffer to the codec, which performs encoding functions. Z buffers are used by specialized controllers, such as ANA4011c, which need to be pre-programmed every time codecs are started. Further, the encoded data is transmitted through the controller and modem to the communication channel. CU initiates the codec chip, and controls the input and output controllers.

CPt decoder is almost identical to CPn encoder, since CPn commands have a fixed length and the encoding and decoding algorithms in all R-S codecs are almost identical. The difference is only in the structure of CPn signals.

The structure of a CPn encoder is shown in Figure 2. Since CPt data block responses are of different length, they have to be recorded in a FIFO memory (First Input First Output) before encoding. At the same time, the number of received symbols is counted in the memory control unit (MCU). If the length of data block exceeds codec’s hardware capabilities, the data flow is conventionally divided into several blocks of a smaller size. Depending on the length of a data block received, the CU sets the required redundancy size based on the given noise resistance value in the given communication system. To speed up the decision-making process and based on the fact that the data redundancy has only a few dozens of levels, it is advisable to select the redundancy size from the pre-programmed CU table. After determining the redundancy size, the CU programs the codec and sends an encoded block to the modem and then to the communication channel through the controller.
Figure 2. Block diagram of CPt encoder.

Figure 3. Block diagram of CPn decoder.

Figure 3 shows the block diagram of CPn decoder. Since the encoding and decoding operations are almost identical, both CPt encoder and CPn decoder circuits are differentiated only by certain control signals and CU operation algorithms. Differences between CU operation algorithms of CPt and CPn are that having determined the length of the encoded block and knowing the CPt table, CU calculates the applied redundancy size and programs the codec accordingly. Another difference is in additional FIFO2 and MCU2. The memory buffer at the output of the decoder is used when CPt blocks of a large length have been divided into smaller blocks. In FIFO2 all data parts are assembled without inclusion of redundancy bites applied for them. After the end of decoding, CPt sends a response to the system output where it processed, for example, by CPn computer.

Since adaptive redundancy calculation algorithms in the form of tables as well as codec chips and memory controlling algorithms are not highly time-consuming, so at CPn and CPns it is advisable to use CUs and MCUs on programmable logic chips. This will significantly minimize and reduce the cost of adaptive codecs for remote telemetry and command systems.

5. Conclusions
The paper considers a solution of an important task to reduce the time of message transmission in remote telemetry and command networks. It is proposed to change the parameters of the noise-resistant code used in each communication session, depending on the amount of data to be transmitted, that is, to perform adaptive encoding. Along with that, some techniques for finding the optimal redundancy of encoded data have been identified; circuitry for encoder and decoder of remote telemetry and command systems have been designed and experimentally tested. Additionally, the paper proposes dependencies of the symbol error probability for a noise-resistant R-S code on the error probability in the communication channel and on the encoding bit rate. And finally, the authors have described techniques and devices realizing R-S algorithms that show an increase of 200-300% in transmission bit rate for an oil pipeline remote telemetry and command network (without reducing the noise resistance level) compared to relevant conventional networks, which do not use the proposed adaptive encoding.

References
[1] Sklyar B 2003 Digital Communication. Theoretical Background and Practical Application (Moscow: Williams Publ)
[2] Samoylov A G and Samoylov S A 2015 Adaptive Programming in Digital Telemetry Systems 
   *Journal* 3 3-6

[3] Al'shraydakh A M and Samoylov A G 2013 Verification of Noise-resistant Encoding Devices 
   *Journal* 1 48-51

[4] Korn G and Korn T 1977 A Mathematical Handbook for Scientists and Engineers (Moscow: 
   Nauka Publ)