ISSN 1068-3712, Russian Electrical Engineering, 2017, Vol. 88, No. 3, pp. 127–129. © Allerton Press, Inc., 2017.
Original Russian Text © N.L. Dodonova, S.I. Khar'kovskii, B.K. Grigorovskiy, A.E. Dubinin, 2017, published in Elektrotekhnika, 2017, No. 3, pp. 31–34.

Decoding Information on the State of Electrotechnical Objects by Using Reed–Solomon Codes

N. L. Dodonova[a], S. I. Khar'kovskii[b], B. K. Grigorovskiy[b], and A. E. Dubinin[b]

[a]*Samara National Research University, Samara, 443086 Russia*
[b]*Samara State Transport University, Samara, 443066 Russia*
e-mail: journal-elektrotechnika@mail.ru

Received February 17, 2017

Abstract—This paper considers the problem of digital information processing by the example of encoding and decoding the control signal of a wireless working cycle correction technology for an autonomous electrical grid. Being subjected to random interferences or intentional attacks, the encoded signal may be decoded incorrectly, causing disruption (change) in the system operation. In this work, we simulate different types of interference. For subsequent software implementation, algorithms are proposed that simulate group and single errors, as well as errors that are combinations of these two types. The effects of errors of different types on the results of decoding the compressed and uncompressed image files are investigated. This paper also proposes some options for representation of graphic files encoded in the software implementation. The files are encoded using the Reed–Solomon codes and are decoded using the Guruswami–Sudan list-decoding algorithm. A comparison between the files obtained after decoding for different types of errors simulated in the process of encoding is presented.

Keywords: management of electrical grids, wireless technologies, digital information processing, data protection, error-control coding, Reed–Solomon codes, Guruswami–Sudan list-decoding algorithm

DOI: 10.3103/S1068371217030063

Electronic devices, such as computers and Smartphones, are widely used in all areas of human activity, including that of electrical grids. The operation of smartgrids is based on digital information processing. Large-scale adoption of digital technologies and the use of information flows to control processes and systems form the basis for development of such grids.

Smartgrids are based on fast-response devices interconnected via a digital network. These devices are capable of correcting (through wireless technologies) their own working cycles, as well as quickly analyzing anomalies in electricity at different geographic scales. Smartgrids enable remote control of the plants using intermittent energy sources (solar panels, wind turbines, etc.) by controlling the load on the electric system and the quality of electric power.

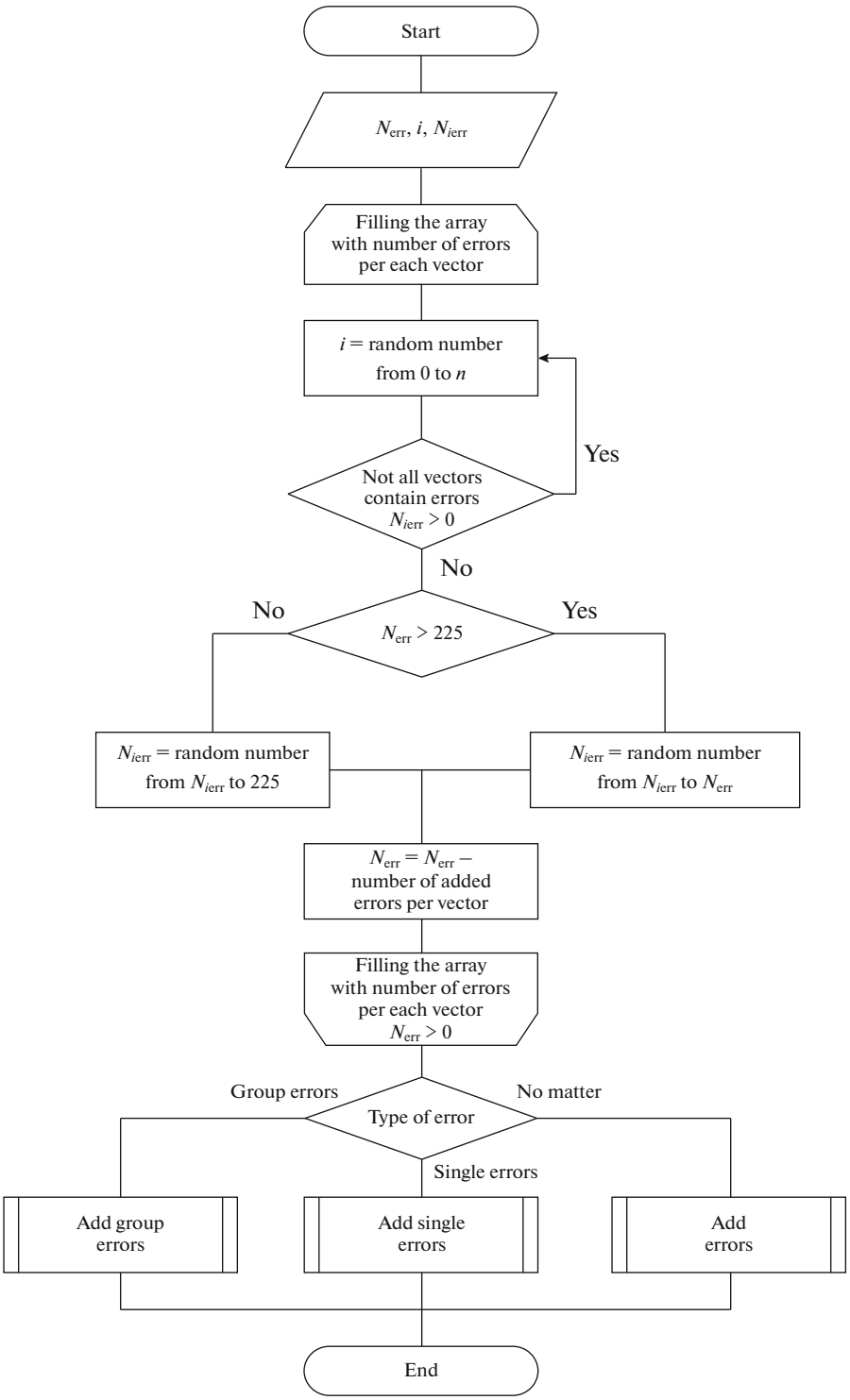
Here, the maintenance of information integrity in the processes of information transmission and storage is a key problem. During transmission, the signal is often exposed to various interferences. Most often, these interferences occur when transmitting an information signal for long distances, particularly, by means of satellite communication [1]. To provide reliable information transmission, special codes are used that find and correct errors caused by interferences (error control codes), with the most popular solutions being the Reed–Solomon codes and various modifications of the Guruswami–Sudan decoding algorithm (in particular, the list-decoding algorithm).

This paper is devoted to investigating the tolerance of an encoded signal aimed at remote control of an autonomous electrical grid for both random interferences and intentional attacks.

Due to the specifics of generating data files of different formats (for example, the use of data compression), in the event of errors, the quality of decoded files can vary significantly. This quality also depends on the arrangement of errors in the code vectors.

To investigate the dependence of the quality of decoded files, we propose a software (Java) implementation of an encoder–decoder device based on the Reed–Solomon codes over the field $GF(2^8)$ [1]. To encode a data vector, we use a generating polynomial of the form $g(x) = (x-\alpha)\left(x-\alpha^2\right)\ldots\left(x-\alpha^{2t}\right)$, α is the maximum primitive element of the Galois field $GF(2^8)$, which equals 254; $t = \frac{1}{2}(n-k)$, where n is the length of a block, which amounts to 255; and k is the user-defined length of the data vector [2].

To decode the data vector, the Guruswami–Sudan list-decoding algorithm [2, 3] is used, which involves two steps. At the first step, the algorithm interpolates the polynomial $Q(x, y)$ of degree $D = \sqrt{nk}$ with respect to x and of degree $L = \sqrt{n/k}$ with respect to y such that the following condition holds: if $(a_0, a_1, \ldots, a_{n-1})$ is the



Error-distribution algorithm.

reference set of the code, then $Q(a_i, y_i) = 0$ for $i = 0.1, ..., n - 1$ and $Q(x, y) \neq 0$.

At the second step, the obtained polynomial $Q(x, y)$ is factored into the divisors of the form $y - p(x)$ such that $\deg p(x) < k$ with $p(a_i) = y_i$ at least for $(n - t)$ values of $i = 0.1, ..., n - 1$.

To simulate signal-transmission interferences, we developed a program in which the user can automatically add errors (that simulate random interferences or international attacks) into an encoded file. For this purpose, the user specifies the number of errors and their type. Here, group errors are those arranged one after another, while single errors are those positioned

Table

Number of errors, % of total number of symbols	Difference from original for group errors, %	Difference from original for single errors, %	Difference from original for combined errors, %
0.2	0.042	0.046	0.040
2.5	0.524	0.566	0.586
4.7	0.987	1.016	1.405

at a certain distance from one another. The third type of error ("no matter") is a combination of the first two types, which means that the file may contain both group and single errors.

Single errors in an encoded file can be interpreted as random interferences, while group and combined errors are interpreted as intentional attacks.

We propose the following algorithm for error distribution over a code vector (see figure).

To generate combined errors (of the "no matter" type), the loop counter is first set to zero; then, a loop is started that adds errors into each vector. This operation is carried out in another loop that adds N_{ierr} errors into the ith vector for n vectors. In this loop, a position for an error is selected randomly. If a given position does not contain errors, then a random number is selected from $GF(2^8)$; otherwise, the position is reselected.

The algorithm for generating single errors differs from the previous one in that the errors (if their number does not exceed 127) are arranged taking into account the neighbor symbols (that should not contain errors), i.e., an extra condition is added for selecting the position of an error. In other words, if the number of errors in a given vector exceeds 127, then the first 127 errors are distributed taking into account the extra condition, while the others are distributed by the algorithm described above.

The algorithm for generating group errors selects only the starting position of a group in the vector in the case in which the starting position of a group is less than or equal to 255; i.e., the group of errors must not fall outside the vector.

Due to the specifics of generating and representing (for encoding) files of different formats (for example, the use of data compression), in the event of errors, the quality of decoded files can vary significantly. The quality of decoded files also depends on the arrangement of errors in the code vectors.

For group errors, the worst case is when the number of such errors in one vector exceeds length $n - \sqrt{nk}$, in which case the Reed–Solomon code fails to correct all these errors; moreover, if the number of group errors exceeds this length considerably, then one or more code vectors may be completely damaged, with their correct decoding being impossible.

For single errors, the probability of the worst case is lower (as compared to group errors). However, in the case of bytewise encoding, even a single uncorrected error can heavily distort the form of a decoded file.

We analyzed the results of decoding for errors of different types. The investigation was carried out on 300 files of various sizes. For encoding, the files were divided into data vectors 195 symbols long. In total, we carried out more than 1500 experiments.

The results of decoding were found by comparison with the original code vector.

Table shows the results of comparison between the transmitted and decoded files.

Thus, the Reed–Solomon codes used to encode a remote-control signal to an autonomous electrical grid yield good results in decoding for both random single errors and deformations of related blocks in the data vector.

REFERENCES

1. Burr, M.T., *Reliability Demands Drive Automation Investments, Public Utilities Fortnightly*, Technology Corridor Department, 2003.
2. McEliece, R.J., The Guruswami-Sudan decoding algorithm for Reed-Solomon codes, *IPN Progress Rep.,* 2003, no. 42-153.
3. Torriti, J., Demand side management for the European supergrid: occupancy variances of European single-person households, *Energy Policy,* 2012, vol. 44, pp. 199–206.
4. Remizov, N.V., Software implementation of Sudan decoder of Reed-Solomon codes, *Inzh. Vestn. Dona,* 2007, vol. 2, no. 2.
5. Monarev, V.A., New high accurate stereoanalysis for raster images, *Prikl. Diskretn. Mat. Prilozhen.,* 2014, no. 7.
6. Shichkin, D.A. and Malykhina, M.P., The way to analyze and apply regularities between color components of RGB model for pattern selecting in images, *Politemat. Setevoi Elektron. Nauch. Zh. Kubansk. Gos. Agrarn. Univ.,* 2013, no. 93.
7. Khar'kovskaya, I.S. and Khar'kovskii, S.I., The way to analyze regularities between text information decoding results and types of mistakes appeared during transition, *Nauka Obrazovanie Transportu,* 2016, no. 1.
8. Khar'kovskaya, I.S. and Khar'kovskii, S.I., The way to apply noise resistant coding for transferring graphical information during using railways, *Programma i tezisy I Mezhdunarodnoi nauchno-praktich. konf. "Innovatsii v sistemakh obespecheniya dvizheniya poezdov"* (Proc. 1st Int. Sci.-Pract. Conf. "Innovations in Trains Motion Supply Systems"), Samara, 2016.

Translated by Yu. Kornienko