BULLETIN of Cherkasy State Technological University

Journal homepage: https://bulletin-chstu.com.ua/en Vol. 29 No. 4. 2024

UDC 004.43
DOI: 10.62660/bcstu/4.2024.62

Algorithms and simulation model for the synchronisation subsystem of the noise-resilient communication system based on permutations

Emil Faure
Doctor of Technical Sciences, Professor
Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
https://orcid.org/0000-0002-2046-481X

Artem Skutskyi*
Postgraduate Student
Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
https://orcid.org/0000-0002-8632-1176

Artem Lavdanskyi
PhD in Technical Sciences, Associate Professor
Cherkasy State Technological University
18006, 460 Shevchenko Blvd., Cherkasy, Ukraine
https://orcid.org/0000-0002-1596-4123

Abstract. In modern data transmission systems, one of the key tasks is to ensure the reliability of communication in conditions of noise. This is especially relevant for channels with high bit error rate, in particular, for radio communication channels with intense natural or artificial noise, which limits the use of traditional error correction methods. The purpose of this work was to develop algorithms for communication using code words of non-separable factorial code, which implies the representation of code words in the form of permutations using simplex binary symmetric communication channel with high bit error probability. To build these algorithms, the method of frame synchronisation of non-separable factorial code, which uses majority and correlation processing of fragments received from communication channel, was taken as a basis. Methods and algorithms for noise-resilient transmission of permutations in communication channels with high bit error probability were investigated. A general scheme of the protocol for organisation of simplex communication was developed. An algorithm for detecting false synchronisations under conditions of high noise level in communication channel was proposed. The effectiveness of the protocol for synchronisation of code words of factorial code was studied, advantages of the used approach were identified and presented. A simulation model of the communication system using a simplex binary symmetric communication channel and the possibility of setting the bit error value in it was developed. The structure of the simulation model and the algorithms of its component blocks were presented. Synchronisation parameters were calculated for bit error probability of 0.4, simulation results based on 10,000 tests were presented, which made it possible to experimentally determine synchronisation algorithm parameters. Simulation modelling was performed and the accuracy of determining the boundaries of synchronisation blocks was estimated based on bit error probability in the range from 0.1 to 0.4. An approach was proposed to reduce the error when determining the boundaries of

Article's History: Received: 12.08.2024; Revised: 20.11.2024; Accepted: 16.12.2024.

Suggested Citation:
Faure, E., Skutskyi, A., & Lavdanskyi, A. (2024). Algorithms and simulation model for the synchronisation subsystem of the noise-resilient communication system based on permutations. *Bulletin of Cherkasy State Technological University*, 29(4), 62-74. doi: 10.62660/bcstu/4.2024.62.

*Corresponding author

Copyright © The Author(s). This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/)

permutations. The obtained results indicate the effectiveness of the proposed solutions, the consistency of theoretical and practical indicators of the synchronisation subsystem, as well as the possibility of using the developed algorithms to implement a three-pass cryptographic protocol based on permutations

Keywords: simplex binary symmetric channel; factorial coding; statistics; protocol; noise

INTRODUCTION

Information transmission is the basis of modern communication systems, providing data exchange in various environments – rom high-speed fiber-optic networks to wireless systems operating in difficult conditions. The performance of systems depends on the ability to maintain the reliability and integrity of information even under the impact of interference, multipath effect, or other factors of communication channel. That is why the development of effective methods for encoding, synchronisation, and protection of transmitted data is one of the key tasks of modern communication technologies.

Modern data transmission technologies face new challenges, which are caused by the need to ensure high reliability of information transmission even in difficult conditions of signal propagation, in particular in environments with high noise intensity. The security of data transmission technology for IoT is critically important, since these devices interact in environments with a large number of potential threats. The vulnerability of IoT devices can cause leaks of confidential information, disruption of systems, or even large-scale cyberattacks. For example, the study by A. Jahangeer *et al.* (2023) notes that the routing protocol for low-power and lossy networks (RPL), specially designed for IoT, is vulnerable to many attacks, such as selective forwarding, blackhole, sybil, wormhole, and sinkhole attacks, which significantly affects network performance. The authors of the study present an overview of attacks on the RPL protocol, as well as a detailed analysis of machine learning and methods for protection against these threats, which highlights the need to improve IoT security.

I. Ahmad *et al.* (2019) and H. Lee & Y.-C. Ko (2021) point out stringent reliability and latency requirements in 5G cellular technologies. Y. Li *et al.* (2022) propose a structure of a digital copy of a device, which through an IoT network using unmanned aerial vehicles (UAVs) can be used as a mobile auxiliary server for edge computing. Such use requires high reliability of data transmission, a secure communication channel, and transmission delays. Modern research, in particular by Y. Sun *et al.* (2021), actively implements optimised data encoding and transmission algorithms, such as byte-parallel configurable cyclic shift (BP-CCS) algorithm in low-density parity-check (LDPC) encoding, which significantly improves data processing efficiency in 5G protocols.

Non-separable factorial codes have been considered and investigated in the study by E. Faure *et al.* (2022). Such codes offer wide opportunities for using permutations as the basis of the transport mechanism in short-packet communication systems. Such systems can be used in IoT systems as discussed in the study by C.D. Ho *et al.* (2021). A similar system is considered by C. Feng *et al.* (2021). The use of permutations in communication is considered by Y. Yang & L. Hanzo (2023). The study compares it with traditional data encapsulation methods. Non-separable factorial codes allow for the implementation of effective mechanism for joint protection of transmitted data from errors occurring in communication channel, as well as ensuring their security from unauthorised access.

As shown by E. Faure *et al.* (2022), code words of non-separable factorial code are formed from a subset of the set of permutations of length M. To ensure their transmission, permutation symbols are encoded by uniform binary code with fixed code combination length. The redundancy of information transmitted by permutations is a feature of such codes. Due to this, non-separable factorial codes, like permutation codes, are able not only to detect, but also to effectively correct errors that occur in communication channel. For example, the study by M. Gao & K.W. Shum (2024) suggests that in the context of combinatorics, permutation codes can be interpreted as directed t-designs and directed t-packings. This connection stems from the fact that permutation code can correct any d-deletions if and only if all subsequences of code words of n-d length are distinct. In addition, the specific structure of permutations creates favourable conditions for frame synchronisation without the need for additional signals, which simplifies the transmission system operation. Under these conditions, sync word length is equal to permutation length, which guarantees signal consistency between the transmitter and the receiver.

Previous studies have demonstrated the effectiveness of non-separable factorial coding even in situations where the probability of bit errors in communication channel remains high. In particular, in the study by B. Stupka (2024) this makes it possible to create: permutation synchronisation algorithms; algorithms for reliable transmission of permutations in communication systems with short packets. At the same time, the developed synchronisation methods imply knowledge of the moment when sync word bits are received, which is not always achievable in conditions of transmission through noisy communication channels. In addition, the integration of synchronisation and protected transmission procedures within one protocol as of 2024 remains insufficiently studied. The purpose of this study was to develop effective algorithms for the operation of the synchronisation subsystem for simplex noise-resilient communication by permutations, as well as to develop a simulation model for their implementation and research.

MATERIALS AND METHODS

There are two systems connected by communication channel. Data transmission through stationary communication channel (*Ch*) occurs in one direction, from one system to another (simplex type of transmission). For better understanding, the names of the systems are indicated according to their target function: the first system is transmission, respectively: *Tx*, the second system is reception, respectively: *Rx*. Information is transmitted from *Tx* through *Ch* to *Rx*. Communication channel distorts the transmitted signal by superimposing noise on the input signal, resulting in bit error probability . Accordingly, *Rx* receives the signal distorted by communication channel. A graphical representation of the communication scheme is shown in Figure 1.

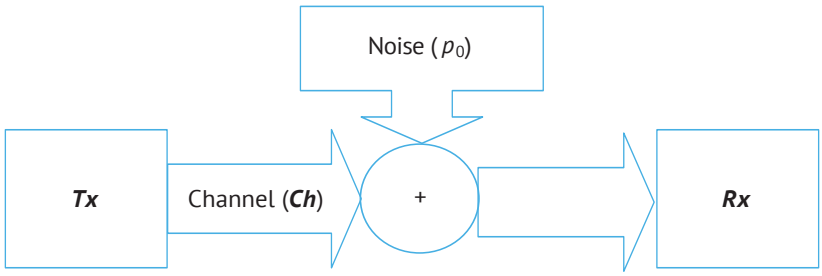

Figure 1. Communication scheme

Source: developed by the authors

Time diagram of the transmission procedure (Fig. 2) shows the components: *Tx*, *Ch*, *Rx*. In addition, there are fragments of information located at the *Tx* output, *Rx* input, *Ch* input and output.

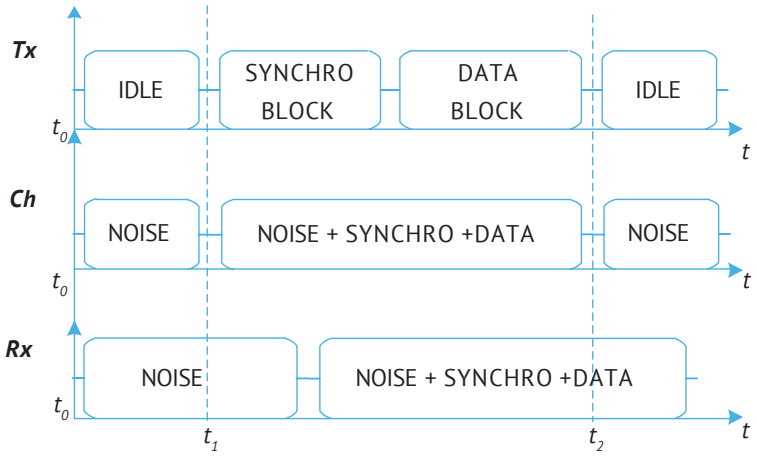

Figure 2. Transmission procedure diagram

Source: developed by the authors

In $[t_0; t_1;]$ time interval *Tx* does not send any data, so the signal level at the input of the channel *Ch* remains constant and zero one. The receiver does not "know" anything about the moment of the start/end of transmission, it constantly processes the input signal. In this case, in $[t_0; t_1;]$ time interval *Rx* receives only noise. Accordingly, during this period of time the probability of bit error in *Rx* is equal to 0.5. In $[t_1; t_2;]$ time interval the transmitter forms a synchronisation block and a data block and sequentially outputs them to the channel.

In the considered transmission procedure, the receiver *Rx* constantly analyses the signal coming from communication channel, since the receiver does not have additional information about the start of transmission from *Tx*, and the *Tx* transmission process can start at any time (asynchronously). Accordingly, the first task for the receiver is to be synchronised with the transmitter to correctly determine the boundaries of transmitted blocks. The second task is to reduce the impact of noise in communication channel on the efficiency of transmission procedures. The solution of these tasks makes it possible to implement a reliable data transmission protocol in communication channels with high bit error probability. In his work, the author B. Stupka (2024) proposes methods for reliable transmission of permutations and demonstrates their effectiveness in the channel with bit error probability $p_0 \leqslant 0.495$. The research by E. Faure *et al.* (2024) is aimed at studying and evaluating the possibility of synchronisation in the channel with $p_0 \leqslant 0.4$. Based on these works, the authors of the study propose the structure of synchronisation block and data block, which is shown in Figure 3.

Figure 3. Structure of synchronisation block and data block

Notes: L_0 – a sync character determined by permutation π that has the maximum value d_{lim} of minimum Hamming distance to all its cyclic shifts; L_{block} – a block of $l_{max\_letters}$ of L_0 sync characters (synchronisation block); W – a word consisting of S characters – cyclic shifts of a sync character L_0: $\{L_1, L_2, \ldots, L_S\}$; W_{block} – a block of l_{max_words} of words W (data block is a frame). Each L_j character is a permutation of symbols of length M. Each symbol is encoded with a binary uniform code. Thus, the length of each symbol is equal to $l_r = \lceil log_2 M \rceil$ bits. Accordingly, the number of bits in a character: $n = M \cdot lr$, the number of bits in synchronisation block L_{block}: $n_L = l_{max\_letters} \cdot M \cdot l_r s$, and the number of bits in data block (frame) W_{block}: $n_w = l_{max\_words} \cdot M \cdot l_r \cdot (n-1)$.
Source: developed by the authors

Permutation synchronisation algorithms with high bit error probability are proposed and researched in the studies by E. Faure *et al.* (2024) and B. Stupka (2024). However, the condition of this study is that the receiver has no knowledge about the beginning of data transmission, and the transmission process can begin at any time. In this case, only the permutation synchronisation algorithm from the study by E. Faure *et al.* (2024) is suitable. This algorithm uses a fixed-size sliding window to ensure a given synchronisation probability, but creates significant redundancy, which increases the transmission overhead. Despite this, its efficiency makes it possible to achieve a theoretical probability of false synchronisation $P_{false\_final} = 2.9 \cdot 10^{-6}$), which makes it promising for applications with high requirements for synchronisation accuracy.

This work uses the frame synchronisation algorithm (synchronisation by characters) without the need to increase the length L_{block} of the synchronisation block. At the same time, the procedures of majority and correlation processing in a sliding window with L_{block} received fragments remain unchanged. A series of cases of false sync detection of the frame synchronisation subsystem is considered as a separate concept. A series of cases for determining false positions of sync character boundaries (a false sync detection series) is a situation when, while searching for the boundaries of a sync character-permutation using a sliding window, the synchronisation subsystem fixes false positions of the boundary for several consecutive shifts of the sliding window. The number of such shifts of the sliding window is the length of the series. To determine the lengths of false sync detection series, a simulation modelling of the frame synchronisation process was carried out. For this purpose, the following parameters of the simulation model were adopted:

- permutation length $M = 8$;
- number of bits to encode the permutation symbol $l_r = 3$;
- bit error rate in communication channel in the absence of transmitter signal $p_{noise} = 0.5$;
- bit error rate in the presence of transmitter signal $p_0 = 0.4$;
- number of tests in the experiment – 10,000.

To ensure the condition of the absence of information about the initial moment of data transmission by the transmitter, the simulation model provides for the formation of a noise block with a length equal to the length of the sliding window of the frame synchronisation subsystem: $L = l_{max\_letters} \cdot M \cdot l_r$ bits. In the conditions of simulation modelling, it is possible to know the start of transmission and the number of bits received by the receiver. Under the condition, the decision on false or true definition of the boundary is formed based on the number of bits received by the l_{rx} receiver, the position of the start of transmission to the synchronisation block (0^{th} bit), the length of sync character fragment and the set of all sync character shifts. Since sync characters in the synchronisation block are arranged sequentially, the sliding window that receives bits is sequential and forms the R refined sequence.

For the next bit reception, the sliding window is shifted by one bit, thus sync character boundary is changed by one. If the frame synchronisation subsystem has determined a boundary that is not equivalent to the calculated boundary from the beginning of synchronisation block transmission, the receiver may set a false boundary of fragments in the synchronisation block. The following expression (1) is the condition for checking the false boundary setting by the frame synchronisation subsystem:

$$S_{false} \rightarrow n_R \neq |l_{rx}|_n, \quad (1)$$

where n_R – the associated cyclic shift of the refined sequence R; l_{rx} – the number of bits received by the receiver Rx. Figure 4 presents a histogram of absolute frequencies of lengths of false sync detection series observed as a result of an experiment with 10,000 tests.

The simulation model, in accordance with the defined algorithm, calculates false sync detection of the synchronisation subsystem under condition (1). If there is a false synchronisation, the run-length counter is incremented. If there is no synchronisation or a true synchronisation occurs, the value of the false synchronisation run-length counter is recorded in the general sample and reset to zero. If the correct sync character

boundary is determined, the current test is completed, and the transition to the next one occurs. From the obtained results of the synchronisation process simulation with the above-defined simulation parameters, the value of the maximum length of the false synchronisation series is $l_{false\_synch} = 4$. The obtained series value makes it possible to form a procedure for increasing the efficiency of input data processing by the receiver *Rx* and successfully receive a frame (block of words) in conditions of high noise intensity.

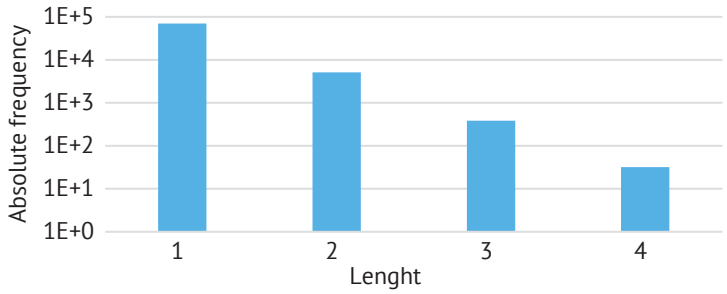

Figure 4. Histogram of absolute frequenciesof false sync detection series
Source: developed by the authors

This procedure is as follows. If the frame synchronisation subsystem run counter exceeds a specified false synchronisation run-length threshold, a decision is made to establish synchronisation. After this, the phase correction process (shifting to the beginning of the boundary of the next fragment of *n* bits) occurs. An attempt to decode the data word begins from the specified position of the next fragment. The algorithm for noise-resilient permutation reception is determined in the study by E. Faure *et al.* (2024). With this procedure, a situation is possible when part of L_0 sync characters of the L_{block} block can be processed by the receiver as characters of the W_{block} block (an error occurs in determining the boundary between L_{block} and W_{block} blocks). The distance from the boundary between L_{block} and W_{block} blocks determined by the synchronisation subsystem to the true boundary is considered to be a numerical value of such an error.

This distance is measured in bits. Based on these expectations, two hypotheses are formed. Hypothesis 1. As a result of performing the frame synchronisation procedure using sync characters, there is an error in determining the boundaries of L_{block} and W_{block} blocks (the error value is not zero). Hypothesis 2. As a result of performing the frame synchronisation procedure using sync characters, there is no error in determining the boundaries of L_{block} and W_{block} blocks (the error value is zero).

RESULTS AND DISCUSSION

Consideration of the structure and algorithms of each component of the simulation model of the data transmission system in accordance with the specified transmission procedure. The structure of the transmitter *Tx* (Fig. 5) presents that:

- the word block W_{block} formation subsystem receives input data in the form of permutation π;
- transmitter settings according to L_0, $l_{max\_letters}$, l_{max_words}, W parameters are set by the user;
- the block L_{block} formation subsystem forms the content of the block in accordance with the specified transmitter parameters;
- the transmission subsystem forms L_{block} and W_{block} blocks according to the set transmission parameters.

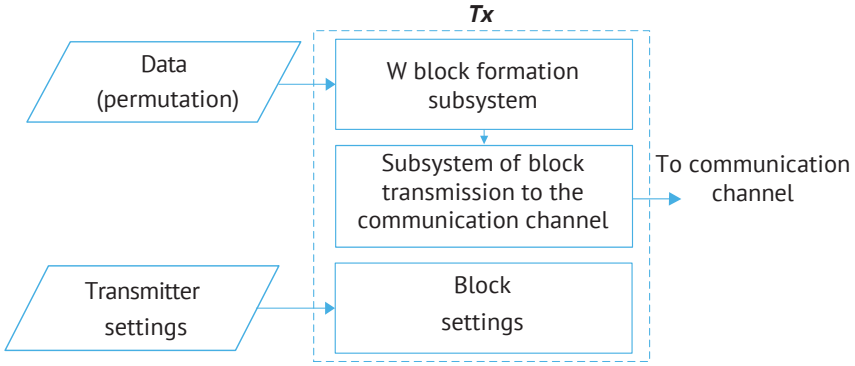

Figure 5. Structure of the transmitter *Tx* system
Source: developed by the authors

The block diagram of the transmitter block operation algorithm (Fig. 6) describes the following sequence of actions:

1. Initialisation. It configures *Tx* based on the entered settings:
   - permutation length *M*;

• number of bits per permutation symbol L_R;
• value L_0 of sync character in the form of a permutation.

Conversion of permutation π into binary representation. Determination of the size of information vectors of synchronisation and data blocks according to $l_{max\_letters}$ and l_{max_words} parameters.

2. Formation of L_{block} and W_{block} blocks from permutations in binary form by combining them into one vector of arrays.

3. Transmission of blocks for output. It transmits bit content of blocks to the transmitter output, to communication channel. Bit-by-bit transmission occurs until the entire content of blocks is transmitted to the output.

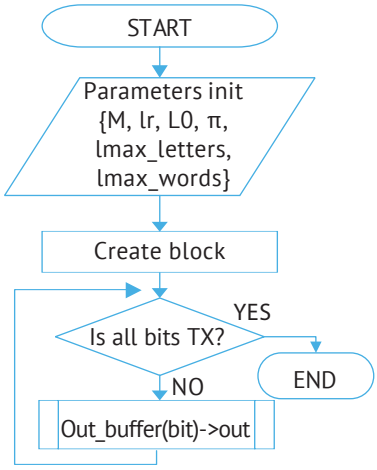

Figure 6. Block diagram of the transmitter *Tx* algorithm
Source: developed by the authors

The structure of the communication channel block *Ch* (Fig. 7) contains:

• input, the channel receives data and transmits it to a noise superposition block;
• noise superposition block that adds generated noise to input data and transmits it to the output;
• noise generation block that generates a distorted signal in accordance with adjustment of false bit rete value set by the user in the form of p_0 value. Generated noise is transmitted to the noise superposition block;
• output that receives data from the noise superposition block and transmits it to the channel output.

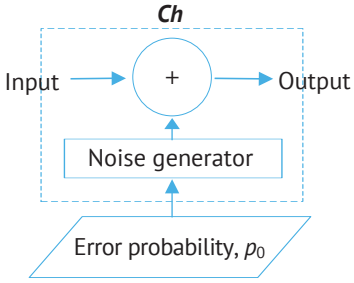

Figure 7. Structure of the communication channel block *Ch*
Source: developed by the authors

The noise superposition block implements the binary symmetric channel (BSC) model. Binary symmetric transmission channel has: an input and an output with an alphabet of binary values that take {1, 0} values (Fig. 8). The output has the probability of false inverted symbol received at p input. The $1-p$ value sets the probability of transmitting the input symbol unchanged, i.e. without distortion.

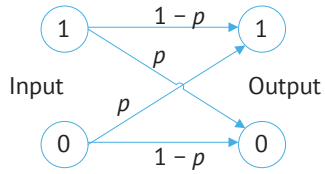

Figure 8. Model of a binary symmetric data transmission channel
Source: developed by the authors

The structure of the receiver block *Rx* (Fig. 9) contains:

• input that receives data from the communication channel system *Ch*;
• input data buffer that accumulates information from the input;
• counter of the number of synchronisation cases that records the length of a series of sync detection of the synchronisation detection subsystem;
• synchronisation detection subsystem that performs data analysis in the input data buffer, the results of the analysis are displayed by increasing or resetting the counter of the number of synchronisation cases;
• phase compensation subsystem that performs phase shift compensation in case the threshold of the synchronisation detection subsystem series is exceeded;
• word processing subsystem that analyses the input data buffer after working out of the phase compensation subsystem;
• output data buffer that stores and displays the decoded information by the word processing subsystem.

The block diagram of the algorithm of the receiver block *Rx* operation is shown in Figure 10.

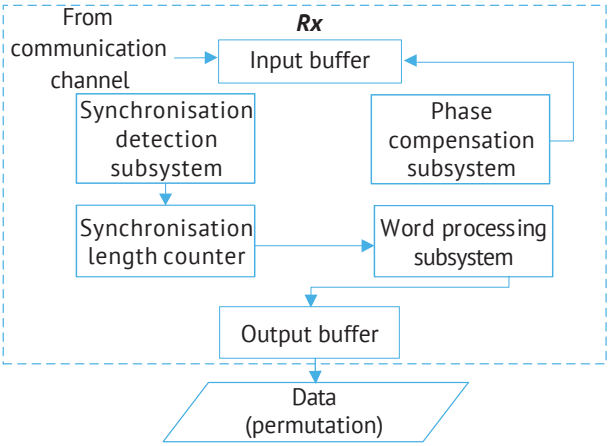

Figure 9. Structure of the receiver block Rx
Source: developed by the authors

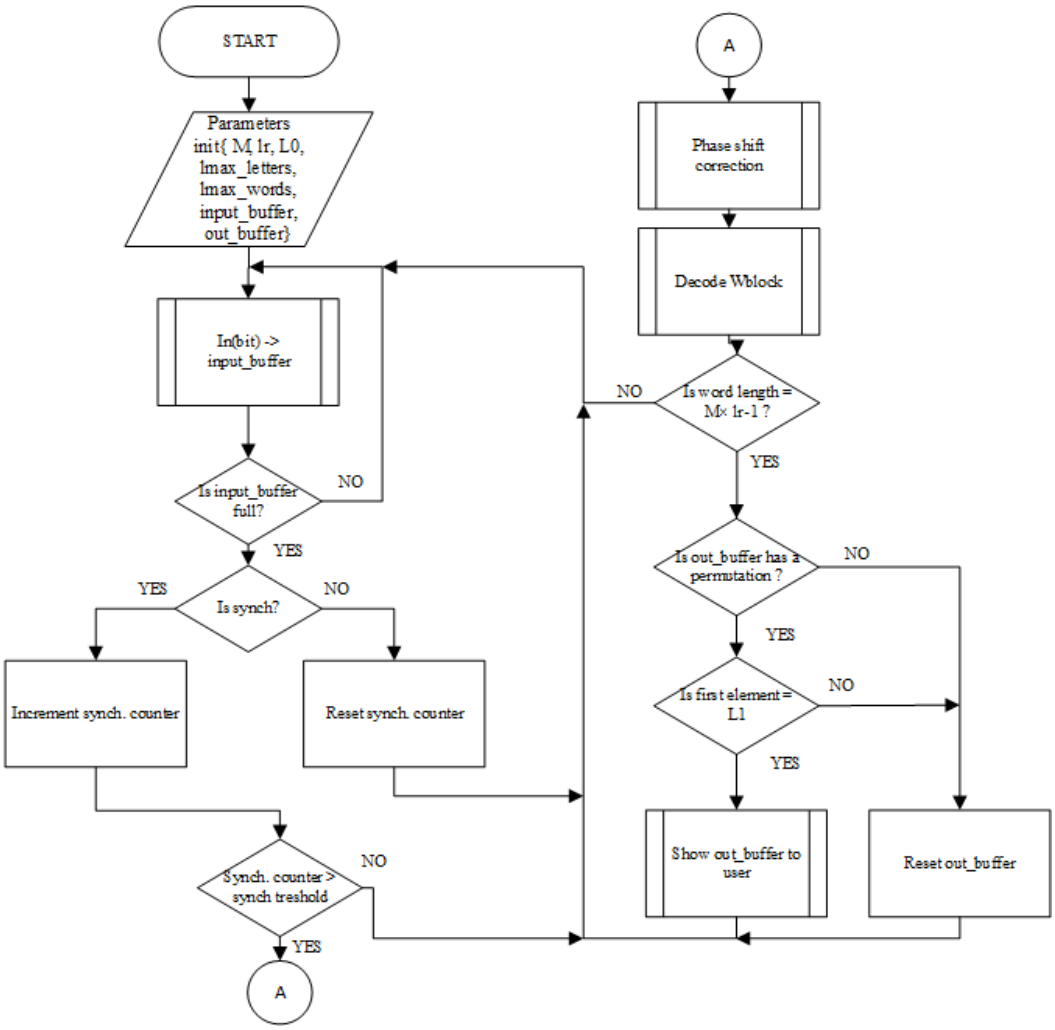

Figure 10. Block diagram of the receiver block Rx operation algorithm
Source: developed by the authors

The receiver block operation algorithm is as follows:

1. Initialisation. It adjusts the receiver according to the settings:

- value of L_0 sync character in the form of a permutation;
- value of the permutation length M;
- number of bits l_r per permutation symbol;
- permutation length n in bits;
- values of all bit and cyclic shifts of L_0 sync character;
- it converts permutation π into a binary representation;
- number of sync characters $l_{max\_letters}$;
- number of words l_{max_words};
- minimum value of the synchronisation series threshold: $l_{synch\_tresh} > l_{false\_synch}$;
- it determines the size of both input and output buffers based on $l_{max\_letters}$ and l_{max_words} parameters.

2. Accumulation of information from the channel. Constant accumulation of bits received at the input of the receiver block and stored in the input data buffer. The buffer has a size equal to the size of synchronisation and data blocks: $l_{rx\_in} = L + W$, bits. The algorithm for recording to the buffer when it is filled can be represented in the form of a queue, first in first out (FIFO). When the buffer is filled with the maximum number of bits l_{rx_in}, the first bit of information by index is deleted. Next bits, starting with index two and ending with l_{rx_in} last bit, are shifted to the left, freeing up space for recording the next bit that comes from communication channel.

3. Synchronisation analysis. Having accumulated l_{rx_in} bits, the receiver analyses a part of the input buffer equal to L bits and forms the refined sequence R. If refined bit sequence is associated with L_0, the synchronisation series counter is incremented. Otherwise, the synchronisation series counter is reset to zero and the transition to receiving the next bit from the input of the receiver block is performed (point 2).

4. Synchronisation series analysis. If the value l_{synch_tresh} of synchronisation series length is exceeded, go to point 5, otherwise receive the next bit (point 2).

5. Phase correction. Based on the obtained data of sync character, the value of the synchronisation phase

shift is determined. For this, the refined sequence R is compared with all cyclic shifts of the sync character. If d_{lim} with a certain shift is present, the number of this j shift is obtained. The number of bits to the boundary of the next $J_{shift} = n - j$ permutation block (fragment) is calculated. If the value of $J_{shift} = 0$, go to point 6, if $J_{shift} \neq 0$, accumulate the number of required J_{shift} (points 2-5).

6. Word block analysis. A data fragment, which is equal to W bits, is selected from the input buffer. The refined sequences for each of $[L_1; L_j]$ characters in the block W_{block} are determined. A comparison with all cyclic shifts of $R \in [L_1; L_j]$ sync character is performed. As a result of the comparison, the identified shift number j for each of refined sequences is obtained. The obtained shift number is recorded to the output buffer. If the word is not recognised after the comparison procedure with L_0 cyclic shifts, clear the output buffer and go to point 2.

7. Word content analysis. The number of entered shifts in the output buffer is counted. When $j = \{1; n-1\}$ shift numbers are accumulated, it is checked whether the numbers are repeated (numbers of word shifts form a permutation). If the numbers are unique, a permuta tion is formed in the buffer, and the contents of the buffer are analysed by index one. If the first number in the output buffer is equal to one (identified with the first cyclic shift of L_1 sync character-permutation), the content of the output buffer is transmitted to the user. A conclusion is formed that the reception of data (transmitted permutation by the transmitter) is completed. If no unique values of j shift numbers have been entered in the buffer or the first number in the output buffer is not equal to one, then clear the output buffer and repeat the reception cycle from point 2.

In the process of testing, the simulation model was experimentally investigated by conducting 1,000 experiments. During the experiment, the data presented in the form of permutation $\pi = (1, 2, 3, \ldots, 23)$ was transmitted for different bit error rate $p_0 = [0.1; 0.4]$ with a step of 0.05. L bits of zero values were transmitted to communication channel, then a synchronisation block L_{block} and a word block W_{block} were formed.

Transmitter system setting parameters:

- permutation length $M = 8$, symbols;
- number of bits per permutation symbol $l_r = 3$, bits;
- value of sync character-permutation $L_0 = (0, 1, 7, 3, 2, 5, 4, 6)$;
- number of sync character blocks $K = 1$;
- number of sync characters in the block $l_{max\_letters} = 75$;
- number of words in the data block $l_{max\_words} = 89$.

Receiver system setting parameters:

- permutation length $M = 8$, symbols;
- number of bits per permutation symbol $l_r = 3$, bits;
- value of sync character-permutation $L_0 = (0, 1, 7, 3, 2, 5, 4, 6)$;
- number of sync character blocks $K = 1$
- number of sync characters in the block $l_{max\_letters} = 75$
- number of words in the data block $l_{max\_words} = 89$.
- minimum value of synchronisation series threshold $l_{synch\_tresh} = 5$.

Based on the results of the experimental study of the simulation model, a sample of the following parameters was formed:

1. The number $N_{frame\_received}$ of data blocks successfully received by the receiver – a situation when the receiver was able to correctly perform synchronisation by characters and decoded the word block – the test was successful.

2. The number N_{frame_lost} of data blocks not received by the receiver – a situation when the receiver was unable to correctly perform synchronisation by characters or correctly performed synchronisation, but incorrectly decoded the word block – the test was not successful.

3. The maximum number of bits from the block L_{block} to the sliding window for analysing the word block W_{block} is $l_{max\_letter\_block}$.

4. The minimum number of bits from the block L_{block} to the sliding window for analysing the word block W_{block} is $l_{min\_letter\_block}$.

5. The number of successfully decoded frames with the maximum number of bits from the block L_{block} to the sliding window for analysing the block W_{block} is $N_{rx\_max\_l}$.

6. The number of successfully decoded frames with the minimum number of bits from the block L_{block} to the block W_{block} is $N_{rx\_min\_l}$.

7. Average transmission time t_μ in seconds.

The results obtained for each $p_0 = [0.1; 0.4]$ with a step of 0.05 as a result of 1,000 tests are given in Table 1.

Table 1. Results of the experiment

BER p_0	0.1	0.15	0.2	0.25	0.3	0.35	0.4
$N_{frame\_received}$	1,000	1,000	1,000	1,000	1,000	1,000	1,000
N_{frame_lost}	0	0	0	0	0	0	0
$l_{max\_letter\_block}$	552	552	552	552	552	552	552
$l_{min\_letter\_block}$	-*	-*	-*	-*	-*	-*	0
$N_{rx\_max\_l}$	1,000	1,000	1,000	1,000	1,000	1,000	988
$N_{rx\_min\_l}$	0	0	0	0	0	0	12
t_μ	0.716	0.706	0.689	0.7	0.7	0.67	0.666

Note: * – value is equal to the maximum number

Source: developed by the authors

To assess the impact of false determination of sync character boundary by the receiver synchronisation subsystem and to check the effectiveness of the receiver operation algorithm, another 1,000 tests were carried out with similar test parameters, the parameter of the minimum value of synchronisation series threshold was set to $l_{synch\_tresh} = -1$ in the receiver block *Rx*. According to the receiver block operation algorithm (Fig. 10), such an indicator turns off the analysis of synchronisation series. The results of the conducted experiment are shown in Table 2.

Table 2. Results of experiments with the turned off mechanism of synchronisation series analysis

BER p_0	0.1	0.15	0.2	0.25	0.3	0.35	0.4
$N_{frame\_received}$	1,000	1,000	1,000	1,000	1,000	1,000	1,000
N_{frame_lost}	0	0	0	0	0	0	0
$l_{max\_letter\_block}$	552	552	552	1,104	1,104	1,104	1,104
$l_{min\_letter\_block}$	-*	-*	-*	552	552	552	552
$N_{rx\_max\_l}$	1,000	1,000	1,000	2	56	317	318
$N_{rx\_min\_l}$	0	0	0	998	944	683	682
t_{μ}	0.68	0.681	0.69	0.736	0.702	0.613	0.488

Note: * – value is equal to the maximum number

Source: developed by the authors

According to the results of the experiment, 1,000 tests were conducted (Table 1). No lost transmitted blocks were recorded during tests, the receiver block reception algorithm with the proposed structure (Fig. 9) successfully received all transmitted fragments through communication channel with $p_0 = [0.1; 0.4]$ with a step of 0.05. Under these conditions, the error in determining the distance between L_{block} and W_{block} blocks did not exceed the value of 552 bits. For $M = 8$ and $l_r = 3$ parameters, this is equal to one word W (phase shift by words). Average synchronisation and decoding time of one frame is 0.666 s with $p_0 = 0.4$. Relative frequency of occurrence of an error in determining the boundary between L_{block} and W_{block} blocks per word for $p_0 = 0.4$ is equal to: $\omega_{word} = \frac{988}{1{,}000} = 0.988$. Relative frequency of occurrence of a zero error in determining the boundary between L_{block} and W_{block} blocks is equal to: $\omega_0 = 1 - \omega_{word} = 0.012$. According to p_0 from 0.1 to 0.35, relative frequency of occurrence of the error in determining the boundary between L_{block} and W_{block} blocks per word is 1.0.

The results of 1,000 tests are shown in Table 2. The algorithm of the receiver block performed the analysis of the synchronisation series length by the l_{synch_tresh} parameter. According to Table 2, no lost transmitted fragments were recorded during tests. The error value for determining the boundary between L_{block} and W_{block} blocks is 1,104 bits, which is equal to $2W$ two words with a p_0 ranging from 0.25 to 0.4. Relative frequencies of error values for determining the boundary between L_{block} and W_{block} blocks are shown in Figure 11. Average synchronisation and decoding time for one test is 0.488 s for $p_0 = 0.4$,

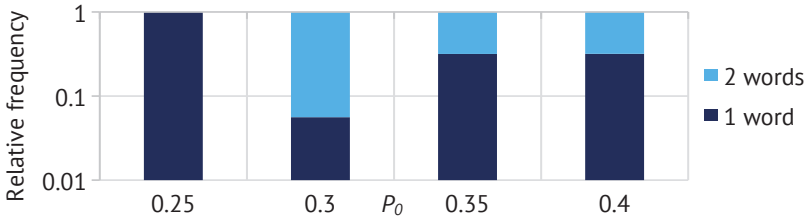

Figure 11. Histogram of relative frequencies of error values for determining the boundary between L_{block} and W_{block} blocks

Source: developed by the authors

The presented simulation model of communication uses binary symmetric channel (BSC) model to simulate the transmission channel. The BSC model can be used for theoretical analysis of data transmission channels, in particular in the tasks of bandwidth calculation, as shown in the study by H. Chen *et al.* (2024), where optimisation using symmetrised KL information is considered. At the same time, the BSC model is a simplified idealisation of real data transmission channels, which has a number of drawbacks in the context of noise simulation:

1. Unrealistic assumption of uniform error distribution. BSC assumes the same error probability for each bit regardless of previous ones. In real channels, errors often have a correlation.

2. Lack of noise type simulation. The BSC model does not take into account physical nature of noise (for example, white Gaussian noise, impulse noise, or interference), which is important in real systems.

3. Limitations of adaptation to the environment. BSC does not take into account variable channel

characteristics, such as dynamic signal-to-noise ratio (SNR), which can change the error probability depending on current conditions. This characteristic is inherent in complex modem systems where information presentation and transmission are implemented using phase and amplitude modulation, where errors depend on signal and noise interaction.

4. Ignoring of packet or symbol losses. In many real channels, not only errors in bit values, but also data loss, which BSC does not simulate, can occur.

5. Non-consideration of spatiotemporal effects. BSC does not take into account the effects of multipath and signal displacement in time, which can significantly affect the quality of transmission.

The study by J. Boiko *et al.* (2022) shows the influence of spatiotemporal effects on the noise resiliece of telecommunication systems with orthogonal frequency-division multiplexing (OFDM) modulation. When implementing the simulation model of the communication system, the phenomenon of signal displacement in time is taken into account by implementing the transmitter in the form of a separate system, as well as transmitting a block of zero bit values with a length of $L = l_{max\_letters} \cdot M \cdot l_r$ into the channel that contains noise after the system passes through communication channel. An alternative to the use of the BSC model implies the following:

1. Binary channel with dependent error probability. To take into account the correlation of errors, it is possible to use models with Markov processes. Thus, V. Vlasenko *et al.* (2023) propose the use of Markov models for quantitative assessment of network reliability, which allows predicting the state of the network and planning preventive measures to ensure its stable operation.

2. Real noise simulation. For a more accurate representation of the environment, it is possible to use channels that take into account the type of noise: a channel with additive white Gaussian noise (AWGN); a channel with impulse noise for cases when interference is short-term and intense. The study by V. Anand Kumar & V. Nandalal (2024), which conducts research and evaluation of the 5G 3GPP cellular network standard, uses several transmission channel models, including both the AWGN channel and the BSC model.

3. Multipath models. For example, Rayleigh or Rice models for simulating multipath effects in wireless networks.

4. Channels with a variable signal-to-noise ratio (SNR).

5. Iterative modelling using realistic simulators.

The overview of the functionality of use of MATLAB software which is shown in the study by D.T. Valentine & B.H. Hahn (2022). The study by J.C. da Silva *et al.* (2021) considers the possibilities of using the NS-3 network simulator to study the LoRAWAN protocol used in IoT. The use of such tools allows for the integration of more complex channel models taking into account real physical effects. To maintain the simplicity and increase the realism of binary symmetric transmission channel, some aspects can be improved by using the Hilbert-Elliot model. H.Q. Ta *et al.* (2022) consider the impact of noise and uncertainty on message throughput using the Gilbert-Elliot model to simulate communication channel. The described approaches will help to increase the accuracy of simulation studies and to ensure greater compliance with real conditions of information transmission.

Word synchronisation and decoding procedure used in the simulation model. In the presence of synchronisation, phase shift compensation occurs only to the boundary of the next sync character fragment, and not to the beginning of the W_{block} block with words, since the receiver does not know where the beginning of the block in the buffer of input information from *Ch* communication channel is. Therefore, having the value of a sync character-permutation, its structure and properties, the receiver block algorithm in the simulation model of communication compensates for the phase shift of the sliding window only until the beginning of the next permutation fragment. When the boundaries of a word block and a character block are incorrectly determined, an error occurs in determining the boundary, which affects the probability of a bit error in the word block. The analysis of the results given in Table 1 with $p_0 = 0.4$ bit error probability in communication channel indicates the presence of an error in determining the boundary of 552 bits. This fragment does not belong to the data block and, accordingly, increases the probability of a bit error in it. This probability can be estimated by expression (2):

$$p_0^* = \frac{l_{max\_letter\_block} \cdot 0.5 + (n_w - l_{max\_letter\_block}) \cdot p_0}{n_w}. \quad (2)$$

For specified model parameters, $p_0^* = 0.401$. The error in determining the boundary occurs due to the fact that the receiver does not know the exact beginning of the block of sync characters-permutations and relies only on information about the structure of information flow. Accordingly, after establishing synchronisation, the system performs phase shift correction to the boundary of the character and assumes that the next block of *W* bits may contain information with words. The reduction in bit error probability when decoding a word depends on accurate determination of the boundary of L_{block} and W_{block} blocks. The use of a mechanism for detecting the synchronisation series threshold exceeding is one of the means for increasing accuracy. The mechanism is represented in the form of a counter showing the length of the series, and in the model algorithm as l_{synch_tresh}.

The effectiveness of the mechanism for detecting the synchronisation series threshold exceeding can be assessed by comparing the results presented in Table 1 and Table 2. Based on the results of two experiments, it is possible to draw a conclusion about the proposed Hypothesis 1 and Hypothesis 2 regarding the synchronisation

series length tracking mechanism. Hypothesis 1 has been confirmed, since after exceeding the threshold for the number of synchronisation cases, phase shift correction, the beginning of analysis and word decoding, the value of the error of determination between L_{block} and W_{block} blocks is not zero. Accordingly, Hypothesis 2 has not been confirmed. However, according to calculated boundary probability of bit error in communication channel for the number of $L_{block} = 1$ and $l_{max\_letters} = 75$ sync character blocks, there are cases where the error in determining the boundary of L_{block} and W_{block} blocks is equal to zero. Table 1 shows the results under conditions of using the mechanism for detecting the synchronisation series threshold exceeding. In this case, the error value is 552 bits. However, as a result of tests, where the mechanism for detecting the synchronisation series threshold exceeding has not been used, the error value for determining the boundary of L_{block} and W_{block} blocks is 1,104 bits (Table 2). Accordingly, the introduction of the mechanism for detecting the synchronisation series threshold exceeding helps to increase the accuracy of determining the boundary of L_{block} and W_{block} blocks. The analysis of synchronisation series helps to reduce the value of the phase shift by words by half and increases the accuracy of word recognition.

Average time taken to decode a word, when using the synchronisation series length tracking mechanism, was 0.666 s with $p_0 = 0.4$. In the absence of the mechanism for detecting the synchronisation series threshold exceeding, this time with the same BER was 0.488 s, which was 26.7% faster. BER in the word block during analysis and decoding increases from 0.401 to 0.402. It is possible to compensate for the error in determining the boundary of L_{block} and W_{block} blocks by performing compensation of the determined boundary after the synchronisation procedure by the value of one word during analysis and decoding by the receiver block. Thus, the error in determining the boundary of sync character and word blocks will decrease from 99.8% to 0.2% under specified model parameters and $p_0 = 0.4$.

The difference between the synchronisation algorithm of E. Faure *et al.* (2024) and the algorithm in the study by B. Stupka (2024) consists in the use of a fixed-size sliding window, where the data received from communication channel is recorded. As a result, the algorithm proposed in the study by E. Faure *et al.* (2024) does not use a dynamic change in values of K and l – the number of blocks (K) of l fragments of length M received from the symbol channel. Therefore, K and l values are constant. They are chosen in such a way as to satisfy the requirements for the probabilities of true and false synchronisation. The length of the sliding window is equal to $K \cdot L \cdot M$ symbols.

Using majority and correlation processing of received fragments, the algorithm of E. Faure *et al.* (2024) makes it possible to achieve a theoretical probability of false synchronisation $P_{false\_final} = 2.9 \cdot 10^{-6}$ for $p_0 = 0.4$, as well as $K = 4$, $l = 85$ and $M = 8$. The presence of significant redundancy is a disadvantage of the frame synchronisation algorithm proposed in the study by E. Faure *et al.* (2024). With an increase in the number of L_{block} blocks having a length of 1,800 bits (for $M = 8$, $l_r = 3$, $l = 75$ parameters) by 4 times, the number of sync character bits in the synchronisation block will be 8,160 bits (for $M = 8$, $l_r = 3$, $K = 4$, $l = 85$ parameters). Such redundancy will result in an increase of overhead by time during transmission/reception. In the study by J.R. Cotrim & J.H. Kleinschmidt (2020), the authors review and evaluate the Long RangeWide Area Network (LoRaWAN) link layer protocol. LoRaWAN is a low-power communication protocol that provides data transmission over distances of up to several kilometers for IoT devices. It operates in radio frequency bands using Long Range (LoRa) technology – spread spectrum modulation, which provides high sensitivity and noise immunity. The study by J.R. Cotrim & J.H. Kleinschmidt (2020) indicates that packet size, spreading factor (SF) and bandwidth are influential factors on the transmission/reception time. The results of the authors' study demonstrate that in the 125 kHz bandwidth of data transmission with a length of 50 bytes and $SF = 7$, the time on air (ToA) is 0.113 s. The calculated channel bandwidth is 3,543.1 bit/s. Using the proposed structure (Fig. 3) and LoRaWAN transmission technology, the number of L_{block} blocks is increased by 4 times and the ToA value will be 2.303 s for the same bandwidth and SF value. The introduction of redundancy leads to a decrease in the "useful" channel bandwidth, since less useful information is transmitted per unit of time, which affects the transmission time. The processing time of received and transmitted information is important when controlling real-time devices or transmitting information in real-time mode.

CONCLUSIONS

The application of algorithms and methods for noise-resilient transmission of permutations through communication channel with high bit error probability was considered and investigated. The structure of the communication system, the synchronisation subsystem, consisting of the synchronisation block L_{block} and the word block W_{block} was solved by conducting simulation modelling of the synchronisation process with 10,000 tests. The maximum length of the series of false determination of sync character boundaries was experimentally determined.

A simulation model of the communication system using a simplex binary symmetric communication channel with $p_0 = 0.4$ was built. The structures and algorithms for the operation of component blocks of the simulation model: transmitter, communication channel, and receiver were presented. The proposed algorithms were tested in the simulation model of communication by conducting 1,000 tests taking into account the mechanism for exceeding the synchronisation series threshold, as well as a simulation modelling of 1,000 transmissions without taking into account the mechanism

for exceeding the synchronisation threshold was conducted in order to assess the impact of the accuracy of determining the boundaries of synchronisation blocks and words. The simulation parameters were presented. The experiments were conducted for a specified range of p_0 = [0.1; 0.4] with a step of 0.05. The simulation modelling has shown that the procedure for taking into account the length of the synchronisation series makes it possible to reduce the error in determining the boundary between L_{block} and W_{block} blocks. The results of the model demonstrate successful information transmission in each of tests with p_0 = 0.4. This indicates the adequacy of the proposed methods and algorithms, which can form the basis for the implementation of the synchronisation subsystem of the noise-resilient communication system based on permutations and be used to implement a three-pass cryptographic protocol.

ACKNOWLEDGEMENTS

This research was funded by the Ministry of Education and Science of Ukraine (Grant No. 0123U100270).

CONFLICT OF INTEREST

None.

REFERENCES

[1] Ahmad, I., Shahabuddin, S., Kumar, T., Okwuibe, J., Gurtov, A., & Ylianttila, M. (2019). Security for 5G and beyond. *IEEE Communications Surveys & Tutorials*, 21(4), 3682-3722. doi: 10.1109/COMST.2019.2916180.

[2] Anand Kumar, V., & Nandalal, V. (2024). A comparative design of 5G communication codes. *International Journal of Communication Systems*, 37(18), article number e5954. doi: 10.1002/dac.5954.

[3] Boiko, J., Pyatin, I., Eromenko, O., & Shayuk, D. (2022). Estimation of the effect of carrier frequency offset on the noise immunity of telecommunications with OFDM. *Measuring and Computing Devices in Technological Processes*, 3, 19-26. doi: 10.31891/2219-9365-2022-71-3-3.

[4] Chen, H., Aminian, G., & Bu, Y. (2024). An algorithm for computing the capacity of symmetrized KL information for discrete channels. In *60th annual Allerton conference on communication, control, and computing* (pp. 1-8). Urbana: IEEE. doi: 10.1109/Allerton63246.2024.10735330.

[5] Cotrim, J.R., & Kleinschmidt, J.H. (2020). LoRaWAN mesh networks: A review and classification of multihop communication. *Sensors*, 20(15), article number 4273. doi: 10.3390/s20154273.

[6] Da Silva, J.C., Flor, D.D.L., de Sousa Junior, V.A., Bezerra, N.S., & de Medeiros, A.A.M. (2021). A survey of LoRaWAN simulation tools in ns-3. *Journal of Communication and Information Systems*, 36(1), 17-30. doi: 10.14209/jcis.2021.2.

[7] Faure, E., Baikenov, A., Skutskyi, A., Faure, D., & Abramkina, O. (2024). Algorithms for reliable permutation transmission protocols in noisy communication channels. In *Cybersecurity providing in information and telecommunication systems II* (pp. 40-48). Kyiv: CPITS.

[8] Faure, E., Shcherba, A., Makhynko, M., Stupka, B., Nikodem, J., & Shevchuk, R. (2022). Permutation-based block code for short packet communication systems. *Sensors*, 22(14), article number 5391. doi: 10.3390/s22145391.

[9] Feng, C., Wang, H.-M., & Poor, H.V. (2021). Reliable and secure short-packet communications. *IEEE Transactions on Wireless Communications*, 21(3), 1913-1926. doi: 10.1109/TWC.2021.3108042.

[10] Gao, M., & Shum, K.W. (2024). Efficient encoding and decoding algorithm for a class of perfect single-deletion-correcting permutation codes. *ArXiv*. doi: 10.48550/arXiv.2411.08258.

[11] Ho, C.D., Nguyen, T.-V., Huynh-The, Th., Nguyen, T.-T., da Costa, D.B., & An, B. (2021). Short-packet communications in wireless-powered cognitive IoT networks: Performance analysis and deep learning evaluation. *IEEE Transactions on Vehicular Technology*, 70(3), 2894-2899. doi: 10.1109/TVT.2021.3061157.

[12] Jahangeer, A., Bazai, S.U., Aslam, S., Marjan, S., Anas, M., & Hashemi, S.H. (2023). A review on the security of IoT networks: From network layer's perspective. *IEEE Access*, 11, 71073-71087. doi: 10.1109/ACCESS.2023.3246180.

[13] Lee, H., & Ko, Y.-C. (2021). Physical layer enhancements for ultra-reliable low-latency communications in 5G new radio systems. *IEEE Communications Standards Magazine*, 5(4), 112-122. doi: 10.1109/MCOMSTD.0001.2100002.

[14] Li, Y., Huynh, D.V., Do-Duy, T., Garcia-Palacios, E., & Duong, T.Q. (2022). Unmanned aerial vehicle-aided edge networks with ultra-reliable low-latency communications: A digital twin approach. *IET Signal Processing*, 16(8), 897-908. doi: 10.1049/sil2.12128.

[15] Stupka, B. (2024). *Methods of reliable information transmission in systems with non-separable factorial coding of data with a high probability of bit error*. (PhD tesis, Cherkasy State Technological University, Cherkasy, Ukraine).

[16] Sun, Y., Li, H., Guo, C., & Wang, D. (2021). A 5G-oriented LDPC encoder based on byte-parallel configurable cyclic shift. In *13th international conference on communication software and networks (ICCSN)* (pp. 17-23). Chongqing: IEEE. doi: 10.1109/ICCSN52437.2021.9463607.

[17] Ta, H.Q., Pham, Q.-V., Ho-Van, K., & Kim, S.W. (2022). Covert communication with noise and channel uncertainties. *Wireless Networks*, 28(1), 161-172. doi: 10.1007/s11276-021-02828-3.

[18] Valentine, D.T., & Hahn, B.H. (2022). *Essential MATLAB for engineers and scientists* (8th ed.). Cambridge: Academic Press. doi: 10.1016/C2021-0-01607-2.

[19] Vlasenko, V., Shchavinskyi, Yu., Zaporozhchenko, M., & Tyshchenko, V. (2023). Analysis of technologies for building a data transmission network with high requirements for information security, reliability and delay. *Connectivity*, 163(3), 8-15. doi: 10.31673/2412-9070.2023.032030.
[20] Yang, Y., & Hanzo, L. (2023). Permutation-based short-packet transmissions improve secure URLLCs in the Internet of things. *IEEE Internet of Things Journal*, 10(12), 11024-11037. doi: 10.1109/JIOT.2023.3243038.

Алгоритми та імітаційна модель підсистеми синхронізації системи завадостійкого інформаційного обміну на основі перестановок

Еміль Фауре

Доктор технічних наук, професор
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
https://orcid.org/0000-0002-2046-481X

Артем Скуцький

Аспірант
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
https://orcid.org/0000-0002-8632-1176

Артем Лавданський

Кандидат технічних наук, доцент
Черкаський державний технологічний університет
18006, бульв. Шевченка, 460, м. Черкаси, Україна
https://orcid.org/0000-0002-1596-4123

Анотація. У сучасних системах передавання даних однією з ключових задач є забезпечення надійності комунікації в умовах завад. Це є особливо актуальним для каналів із високою ймовірністю бітових помилок, зокрема, для каналів радіозв'язку з інтенсивними природними чи штучними шумами, що обмежує використання традиційних методів корекції помилок. Метою цієї роботи була розробка алгоритмів інформаційного обміну кодовими словами нероздільного факторіального коду, що передбачає представлення кодових слів у вигляді перестановок, симплексним двійковим симетричним каналом зв'язку з високою імовірністю бітової помилки. Для побудови цих алгоритмів за основу взято метод циклової синхронізації нероздільного факторіального коду, який використовує мажоритарну та кореляційну обробку прийнятих з каналу зв'язку фрагментів. Досліджено методи та алгоритми завадостійкого передавання перестановок у каналах зв'язку з високою ймовірністю бітової помилки. Розроблено загальну схему протоколу організації симплексного інформаційного обміну. Запропоновано алгоритм детектування хибних синхронізацій для умов високого рівня шуму в каналі зв'язку. Досліджено ефективність протоколу синхронізації кодових слів факторіального коду, виявлено та представлено переваги використаного підходу. Розроблено імітаційну модель системи інформаційного обміну симплексним двійковим симетричним каналом зв'язку та можливістю задання значення бітової помилки в ньому. Наведено структуру імітаційної моделі та алгоритми роботи її складових блоків. Виконано розрахунок параметрів синхронізації для ймовірності бітової помилки 0,4, представлено результати моделювання за 10 000 випробувань, що дозволило експериментальним шляхом визначити параметри алгоритму синхронізації. Виконано імітаційне моделювання та отримано оцінку точності визначення меж блоків синхронізації за ймовірності бітової помилки в діапазоні від 0,1 до 0,4. Запропоновано підхід, що дозволяє зменшити похибку під час визначення меж перестановок. Отримані результати свідчать про ефективність запропонованих рішень, узгодженість теоретичних і практичних показників роботи підсистеми синхронізації, а також про можливість використання розроблених алгоритмів для реалізації трьохетапного криптографічного протоколу на основі перестановок

Ключові слова: симплексний двійковий симетричний канал; факторіальне кодування; статистика; протокол; шум