Research of the effectiveness of frame synchronization methods for creating markerless telecommunication systems for exchanging short packets

Alimzhan Baikenov[1,†], Emil Faure[2,†], Sakhybay Tynymbayev[3,†], Olga Abramkina[3,*,†] and Artem Skutskyi[2, †]

[1]Almaty University of Power Engineering and Telecommunications name after Gumarbek Daukeev, Baitursynov 126/1, 050013, Almaty, Kazakhstan

[2] Cherkasy State Technological University, Shevchenko Blvd. 460, 18006, Cherkasy, Ukraine

[3] International Information Technology University, 34/1 Manas St., Almaty, 050000, Kazakhstan

Abstract

This paper investigates synchronization techniques for developing markerless telecommunication systems. In such systems, explicit markers are eliminated, which minimizes the amount of overhead. Markerless systems outperform traditional systems in tasks requiring high response speed and low latency. One of the key components of such systems is frame synchronization, which ensures correct identification and decoding of data packets at the receiving end without using explicit markers indicating the beginning or end of a packet. The study focuses on the effectiveness of frame synchronization techniques, since the accuracy and reliability of these techniques significantly affect the overall performance and stability of the system. Traditional synchronization techniques such as time synchronization, block counters, and checksums are analyzed. In addition, a new factorial coding technique is proposed that eliminates the need for explicit markers, thereby reducing the overhead and improving the channel efficiency. The results show that factorial coding improves synchronization accuracy, reduces latency, and provides better resilience to interference, making it a promising approach for next-generation markerless telecommunication systems.

Keywords

Markerless telecommunication systems, frame synchronization, efficiency, bit error rate, factorial coding

1. Introduction

Modern telecommunication systems are characterized by growing requirements for data transfer speed, reliability and efficiency of network resource use. In the context of increasing traffic volumes and the diversity of transmitted data, the task of developing methods that ensure fast and reliable information transfer with minimal delays is becoming especially urgent [1,2,3].

These methods are a combination of various technologies and approaches aimed at optimizing data transfer processes in telecommunication systems [4,5,6].

The main methods that are most often used include frame synchronization [7,8,9], adaptive coding and modulation [10, 11, 12], error correction methods [13,14], protocols with low response time [15,16,17]. When comparing these methods for ensuring fast and reliable data transmission with minimal delays in telecommunication systems, it becomes clear that each of them has its own unique advantages and disadvantages that affect their effectiveness in different usage scenarios [18]. Frame synchronization minimizes the amount of service information, which allows for an increase in the data transfer rate, but it can be sensitive to interference, which makes it difficult to operate in unstable networks. Adaptive coding and modulation, on the contrary, flexibly adapt to channel

DTESI 2024: 9th International Conference on Digital Technologies in Education, Science and Industry, October 16–17, 2024, Almaty, Kazakhstan

* Corresponding author.

† These authors contributed equally.

a.baikenov@aues.kz (A. Baikenov); e.faure@chdtu.edu.ua (E.Faure); s.tynym@gmail.com (S. Tynymbaev); olga.manank@gmail.com (O. Abramkina); a.b.skutskyi.asp21@chdtu.edu.ua (A. Skutskyi)

0000-0002-6490-3159 (A. Baikenov); 0000-0002-2046-481X (E.Faure); 0000-0002-9326-9476 (S. Tynymbayev); 0000-0003-0137-1252 (O. Abramkina); 0000-0002-8632-1176 (A. Skutskyi)

conditions, providing a balance between speed and reliability, but require more complex control algorithms and can cause delays due to constant monitoring of the channel state. Foreground error correction (FEC) methods increase the reliability of data transmission due to the ability to detect and correct errors at the receiving end, but at the same time increase the amount of transmitted data by adding redundancy, which reduces the efficiency of bandwidth use. Low response time (RTT) protocols, on the other hand, are optimized for fast exchange of short messages, which minimizes response time and improves real-time performance, but may be less effective in high-noise environments than other methods that provide greater immunity to distortion.

It is worth noting that these methods are based on the principle of using explicit markers to indicate the beginning and end of packets, error correction and other mechanisms that are most often used in traditional telecommunication systems. These systems were developed to ensure reliable and predictable data transmission in conditions of limited bandwidth and exposure of communication channels to various types of interference. Particular attention is paid to frame synchronization, since in traditional systems it ensures correct recognition of data packet boundaries. This is especially important for preventing errors associated with the loss or distortion of markers, which could lead to misinterpretation of data on the receiving side. Frame synchronization also helps traditional systems cope with noise and interference in the communication channel. Frame-level synchronization allows the system to quickly restore synchronization after errors occur, which improves the overall reliability of data transmission [19,20,21]. Many standardized communication protocols, such as Ethernet, GSM or Wi-Fi, use frame synchronization methods to ensure compatibility and reliability. These protocols are designed to operate in a variety of conditions and include time-tested methods such as the use of markers and checksums [22]. Frame synchronization in traditional systems is usually well documented and supported by a wide range of hardware and software. This simplifies the development and implementation of telecommunications systems, and ensures compatibility between different devices and networks.

However, if we consider traditional systems, there is a high probability of errors, interference and data loss.

The article proposes to study code synchronization methods for creating markerless systems. Such systems do not have explicit markers, which allows minimizing the amount of service information. Markerless systems outperform traditional ones in tasks that require high efficiency and low delays. One of the key components of such systems is frame synchronization, which ensures correct recognition and decoding of data packets on the receiving side without using explicit markers indicating the beginning and end of the packet.

The study of the efficiency of frame synchronization methods plays an important role in the creation of markerless telecommunication systems, since the overall performance and stability of the system depend on the accuracy and reliability of these methods. In this paper, the frame synchronization efficiency indicator is considered. Frame synchronization efficiency is understood as the ability of the system to correctly recognize the beginning and end of data frames in the information flow. Traditional frame synchronization methods, such as time synchronization, block counters and the use of checksums, are studied. A method with factorial coding is proposed that does not require explicit markers, which will reduce the amount of service information and increase the efficiency of channel use.

2. Methodology

The study of the efficiency of frame synchronization methods for markerless telecommunication systems was carried out in several stages. The main attention was paid to the comparative analysis of traditional frame synchronization methods, such as time synchronization, block counters and the use of checksums and the factorial coding method.

At the first stage, key performance indicators of frame synchronization methods were determined, such as the noise level in the communication channel, bandwidth, and delays. The ability to dynamically change these parameters during experiments was also implemented to assess the stability of the methods to various transmission conditions.

At the second stage, modeling and data collection were carried out according to the selected parameters. The experimental setup includes software for modeling a telecommunication system and physical equipment, including computers and network devices connected through a configured switch. The software imitated data transmission between network nodes using the developed frame synchronization algorithm and factorial coding in the MATHLAB environment.

The third stage includes the analysis of data transmission results and the construction of an efficiency graph for each method.

3. Experimentation

3.1. Time synchronization

Time synchronization ensures that time is coordinated between network nodes so that all participants in the data transfer process have the same time scale. The choice of parameters is shown in Figure 1, which includes the number of sending data blocks, the size of each data block, and the time interval between data blocks per second.

```
>>
numBlocks = 5;
blockSize = 100;
interval = 1;
```

Figure 1: Determination of experimental parameters.

The principle of the method is important for preventing data transmission collisions and reducing errors. In simulation, this method shows high efficiency under stable network conditions, but its performance noticeably decreased with increasing time delays in the communication channel (Figure 2).

```
Columns 1 through 24

 251    44    66   101    19   174   103   250   103   158    40    97    41

Columns 25 through 48

 108    92   142   189   108   110    32     7    74    81   167   244   238

Columns 49 through 72

 210    45    42   170   228   132   179    40   243   138   173    10   206

Columns 73 through 96

 167   237    42   234   202   147   112    66   191    59    17   195   171

Columns 97 through 100

 242   113    16   221
```

Figure 2: Output of results.

The bit error dependence is shown in Figure 3.

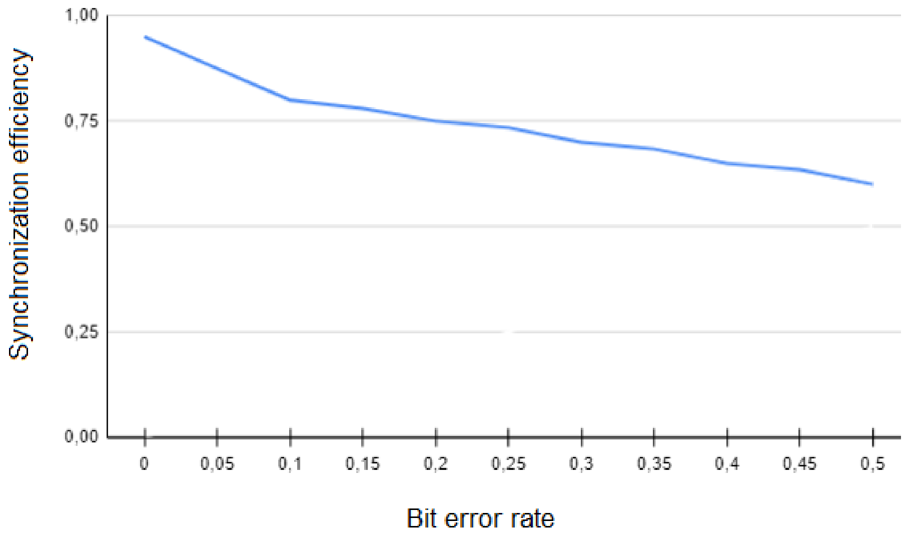

Figure 3: Time synchronization method.

3.2. Block counters

The "block counters" method allows tracking the serial numbers of transmitted data blocks. This approach ensures reliable synchronization even in conditions of significant interference, but requires complex algorithms for processing erroneous or lost blocks.

Here, the main parameters for data generation are set: the number of blocks (numBlocks) and the size of each block in elements (blockSize). These parameters determine the volume and structure of data for the experiment (Figure 4).

```
>>
numBlocks = 5;
blockSize = 100;
```

Figure 4: Defining the Experiment Parameters.

The generated data block with a counter is added to the general array “allDataWithCounters”. After generating and adding all data blocks to the general array, the data is sorted by the first column, which contains the block counters, using the “sortrows” function (Figure 5).

```
sortedData = sortrows(allDataWithCounters, 1);

% Вывод отсортированных данных
disp('Отсортированные данные по номерам блоков:');
for i = 1:numBlocks
    disp(['Блок ', num2str(sortedData(i, 1)), ': ', num2str(sortedData(i, 2:end))]);
end
```

Figure 5: Sorting and outputting sorted data.

This allows the original sequence of blocks to be restored, even if they were “received” in a different order.

The code then outputs the sorted data blocks, showing their counters and contents, allowing the synchronization result to be visually seen (Figure 6).

```
Отсортированные данные по номерам блоков:
Блок 1: 207  231   33  232  161   25   71  139
Блок 2: 42  202   80  135   43  153   67  167
Блок 3: 164   97  207  136   90  239  223  140
Блок 4: 16  174   11   19  133   25  208  208
Блок 5: 108   24  153  120  177  178  163    9
```

Figure 6: Output of results.

The bit error dependence is shown in Figure 7.

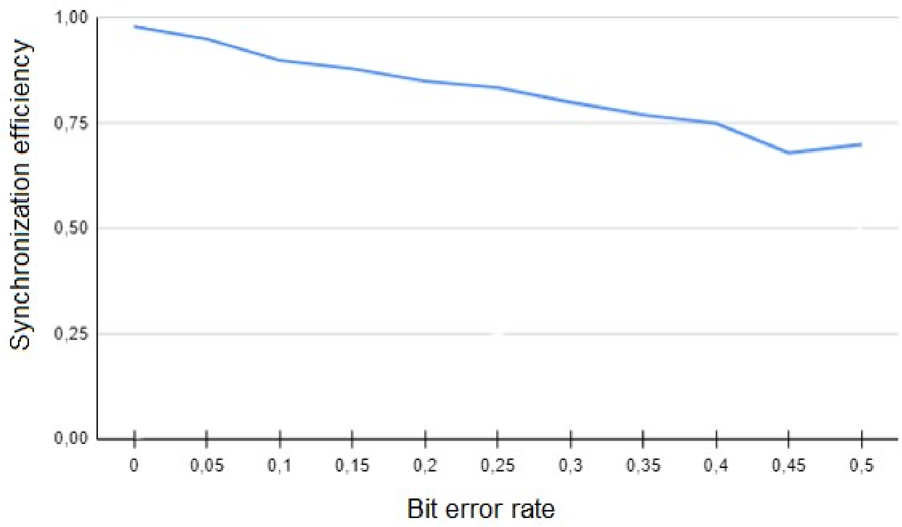

Figure 7: Block counter method.

3.3. Using checksums

In the synchronization method, "checksums" are used to confirm the integrity of blocks and their correct order [22,23]. The choice of parameters is shown in Figure 8.

```
>> % Параметры эксперимента
numBlocks = 5; % Количество блоков данных для отправки
blockSize = 100; % Размер каждого блока данных
```

Figure 8: Determining the Experiment Parameters.

This method showed high efficiency in detecting and correcting errors, but its performance depended on the checksum calculation algorithm. Initialization of the array for storing data blocks with checksums is shown in Figure 9.

```
% Инициализация массива для хранения всех
allDataWithChecksums = [];
```

Figure 9: Initializing an array for storing data blocks with checksums.

Displaying all generated data blocks together with their checksums, simulating the process of their “sending” (Figure 10).

```
% Проверка каждого блока данных на ошибки, используя контрольные суммы
for i = 1:numBlocks
    receivedBlock = allDataWithChecksums(i, :);
    receivedChecksum = receivedBlock(end);
    dataBlock = receivedBlock(1:end-1);

    % Пересчет контрольной суммы
    recalculatedChecksum = mod(sum(dataBlock), 256);

    % Проверка контрольной суммы
    if receivedChecksum == recalculatedChecksum
        disp(['Блок ', num2str(i), ' принят без ошибок.']);
    else
        disp(['Ошибка в блоке ', num2str(i), '!']);
    end
end
```

Figure 10: Receiving and checking data blocks for errors.

When diagnosing network or file problems, checksums can help determine whether an error occurred during data transmission or not (Figure 11).

```
  Columns 97 through 101

    48   126    37    14   236
   119   165     6   215   163
    18    22   204   241    71
   142    80    42   159   234
   243   113    15   221     3

Прием и проверка блоков данных на ошибки:
Блок 1 принят без ошибок.
Блок 2 принят без ошибок.
Блок 3 принят без ошибок.
```

Figure 11: Output of results.

The bit error dependence is shown in Figure 12.

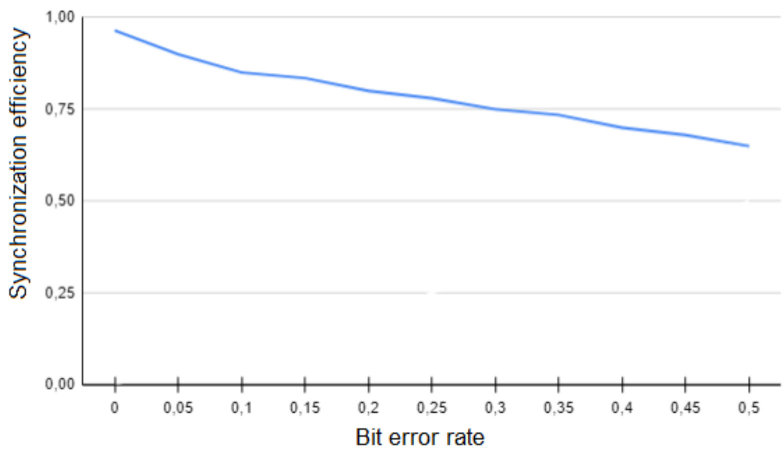

Figure 12: Checksum method.

3.4. Frame synchronization of inseparable factorial code

Inseparable factorial coding telecommunication systems [24] use non-standard and redundant frame structures that do not provide a separate SFD field and can perform a transport function for transmitting short packets [25]. Since the lengths of all code words in an inseparable factorial code are equal, frame synchronization methods use a permutation of elements π and M. The method proposed in [24] complements the processing of binary symbols obtained from the communication channel using a majority gate or n -bit majority scheme with correlation processing [26].

In this paper, the frame synchronization system uses a permutation π, which is a sequence of numbers in a set $\{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\}$. A fixed-length binary code is used to encode each element in this set with bits $l_r = \lceil \log_2 M \rceil = 4$, $(n = M \cdot l_r = 64)$ as shown in Table I[24].

Table 1
Encoding scheme of permutation elements

Decimal notation	Binary notation	Decimal notation	Binary notation
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	12	1101
6	0110	14	1110
7	0111	15	1111

According to [24], the criterion for choosing a synchronization word is the maximum value of the minimum Hamming distance between the binary representation of the permutation and each of its circular shifts. The correct synchronization probability [24] is:

(1)

The probability of false synchronization [26] is:

$$P_{false}\left(n, d_{lim}, p_0, l, K\right) = \quad (2)$$

$$= \sum_{j=1}^{n-1} \left(\sum_{v=d_{ij}-d_{lim}}^{d_{ij}} C_{d_{ij}}^{v} \sum_{w=0}^{v-d_{ij}+d_{lim}} \left(\begin{matrix} C_{n-d_{ij}}^{w} \left(p_0^*\right)^{v+w} \times \\ \times \left(1-p_0^*\right)^{n-v-w} \end{matrix} \right) \right)^K .$$

The method of frame synchronization of an inseparable factorial code with a bit error probability close to 0.5 uses a comprehensive approach, including both theoretical modeling and practical implementation in the MATHLAB programming language. The initial data of the experiment are presented in Figure 13.

```
>> N = 1000; % Количество блоков
S = 256; % Размер блока в битах
P_e = 0.49; % Вероятность битовой ошибки
   % Генерация блоков данных
   dataBlocks = randi([0, 1], [N, S]);
```

Figure 13: Initialization of the experiment parameters and generation of data blocks.

The probability of bit error (“P_e”) is set at 0.49 to get closer to the critical value of 0.5.

These parameters were chosen based on the need to clearly demonstrate the operation of the developed methods under high noise conditions (Figure 14).

```
% Проверка ошибок с помощью контрольной суммы
errorsDetected = 0;
for i = 1:N
    originalChecksum = mod(sum(dataBlocks(i, :) + (1:S)), 256);
    receivedChecksum = mod(sum(receivedBlocks(i, :) + (1:S)), 256);
    if originalChecksum ~= receivedChecksum
        errorsDetected = errorsDetected + 1;
    end
```

Figure 14: Error checking using a checksum.

For each data block, a checksum is calculated, which includes the sum of the values of the data block bits and their ordinal numbers, taken modulo 256. This allows for increased reliability of error detection by taking into account not only the states of the bits, but also their positions in the block.

The checksums are compared for the original and received data blocks. A discrepancy between the checksums indicates the presence of errors, after which the counter of detected errors “errorsDetected” is incremented (Figure 15).

```
fprintf('Обнаружено ошибок: %d из %d\n', errorsDetected, N);
Обнаружено ошибок: 964 из 1000
```

Figure 15: Results output.

At the end, the total number of detected errors relative to the total number of sent data blocks is output. This gives an idea of the effectiveness of the used checksum method under high probability conditions.

Assuming that there is noise in the communication channel, which affects the probability of an error in the transmission of individual bits, we set the task of determining the probability of successful synchronization of data blocks for different noise levels.

The key parameters for the simulation were:

The probability of a bit error in the channel (P_e), varied from 0 to 0.5;

The size of the data block (S), equal to 256 bits;

The number of data blocks (N), equal to 1000.

Based on the results obtained, it was found that with an increase in the probability of a bit error, the probability of successful synchronization of a data block decreases significantly. The graph generated by the code (Figure 16) demonstrated an inversely proportional dependence of the probability of successful synchronization on the probability of a bit error.

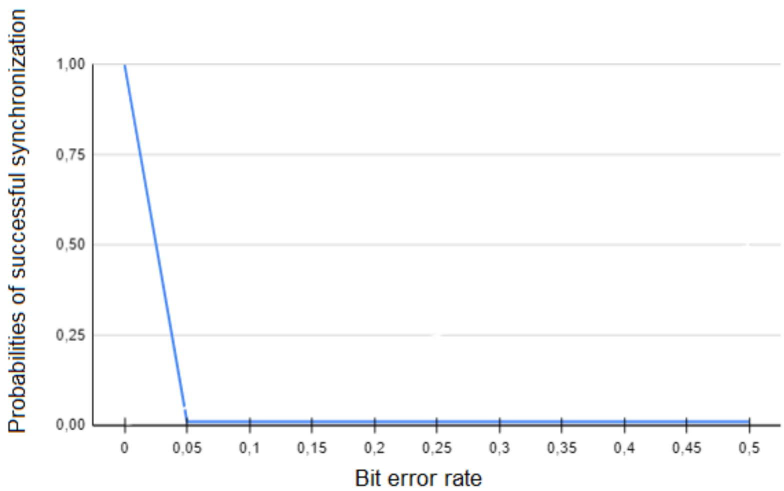

Figure 16: Output of results.

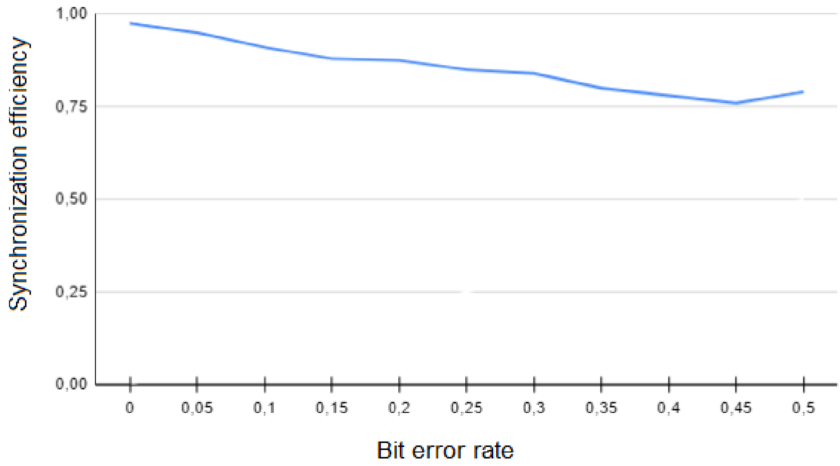

Figure 17: Factorial coding method.

In particular, when “P_e” was equal to 0.1, the probability of successful synchronization was quite high, but when “P_e” approached 0.5, the probability of successful synchronization tended to zero.

4. Analysis of results

After performing a series of experiments for each synchronization method, a comparison of their efficiency was carried out (Figure 18).

In this paper, efficiency is considered as the probabilities of correct and false synchronization.

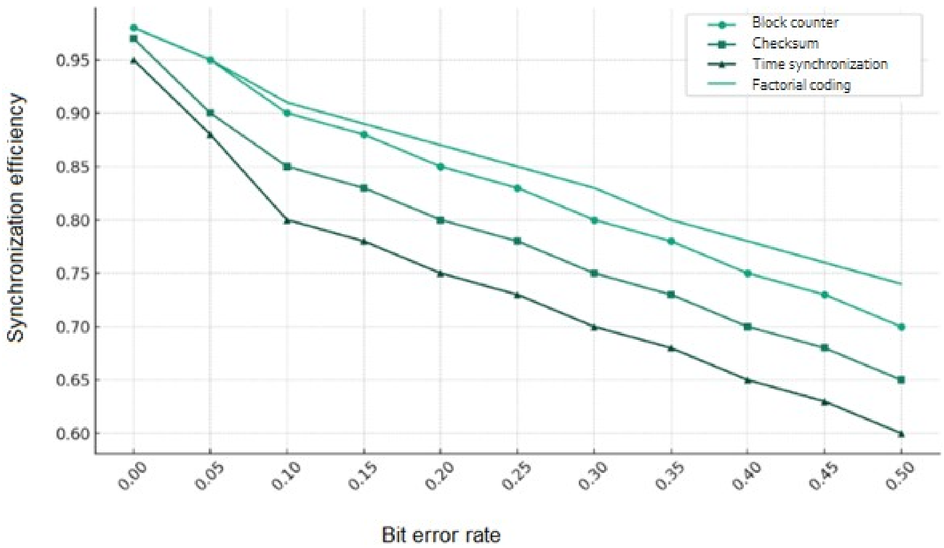

Figure 18: Performance Graph.

It was found that the checksum method showed the highest performance under conditions with a bit error probability of up to 0.3. However, with a further increase in "P_e", its performance decreased faster than that of the block counter method.

The time synchronization method turned out to be the least effective due to its high sensitivity to changes in data transmission delays, which was especially noticeable at high "P_e".

Block counters demonstrated the best overall error tolerance under conditions of high error probability, maintaining relatively high performance even at "P_e" close to 0.5.

5. Conclusion

In this paper, several frame synchronization methods for creating markerless telecommunication systems were investigated and analyzed. Traditional methods, such as time synchronization, block counters and the use of checksums, have demonstrated their reliability in conditions with a moderate level of interference and errors.

The experiments and their analysis not only revealed the most effective synchronization methods under conditions of high bit error probability, but also opened the way for future research aimed at further improving the efficiency and reliability of data transmission.

In the context of modern tasks that require high efficiency and minimal delays, traditional frame synchronization methods may not be effective enough. Markerless systems, which do not use explicit markers, offer significant advantages in these conditions.

The main contribution of this work is the development and proposal of a factorial coding method for frame synchronization in markerless systems. This method allows minimizing the amount of service information, reducing redundancy, and allows transmitting more useful data, which is especially important for high-speed communication systems. Thus, the proposed factorial coding method is promising for tokenless communication systems and can significantly improve their performance, especially in scenarios requiring high data rate and low latency.

Acknowledgements

This research has been/was/is funded by the Science Committee of the Ministry of Education and Science of the Republic of Kazakhstan "AP23489168 Methods and protocols for secure information exchange based on factorial coding of data and transformations in finite matrix fields".

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] L. Jun, L. Feng, A. Nirwan, Monitoring and analyzing big traffic data of a large-scale cellular network with hadoop, IEEE Netw., vol. 28, no. 4, 2014, pp. 32-39.

[2] S. Uthayasankar, K. M. Mustafa, I. Zahir, W. Vishanth, Critical analysis of big data challenges and analytical methods, J. Bus. Res., vol. 70, 2017, pp. 263-286. doi: 10.1016/j.jbusres.2016.08.001

[3] Z. Ayan, B. Alimzhan, M. Olga, Z. Timur, and Z.Toktalyk, Quality of service management in telecommunication network using machine learning technique, Indonesian J. of Electr. Eng. and Comput. Sci., vol. 32, no. 2, pp. 1022–1030, 2023. doi: 10.11591/ijeecs.v32.i2.pp1022-1030.

[4] Y. Wang, S. Ram, F. Currim, E. Dantas, L.A. Sabóia, A big data approach for smart transportation management on bus network. In Proceedings of the 2016 IEEE International Smart Cities Conference (ISC2), Trento, Italy, 12–15 September 2016; pp. 1–6.

[5] S.B. Mukhanov, A.A. Aldanazar, A.M. Uatbayeva, A.Ye. Alimbekov, G.S. Marat, Competitive learning in neural networks, Int. J. of Inf.and Commun. Tech. vol. 1, no. 3, 2023. doi: 10.54309/ijict.2020.3.3.011.

[6] T. Velyamov, A. Kim, O. Manankova, Modification of the Danzig-Wolf Decomposition Method for Building Hierarchical Intelligent Systems, Int. J. of Adv. Comput. Sci. and Appl., vol. 15, no. 7, pp. 1160–1167, 2024. Doi: 10.14569/IJACSA.2024.01507113.

[7] B.F. Bregman, V.V. Kvashennikov, Method of code frame synchronization of messages, RF. Patent 2 459 366, 2006.

[8] B. Bloessl, F. Dressler, mSync: Physical Layer Frame Synchronization without Preamble Symbols, in IEEE Transactions on Mobile Computing, vol. 17, no. 10, pp. 2321-2333, 2018. doi: 10.1109/TMC.2018.2808968.

[9] J. Hamkins, Frame synchronization without attached sync markers, in 2011 Aerospace Conference, pp. 1-7, 2011. doi: 10.1109/AERO.2011.5747327.
[10] D. Tarchi, G. Corazza, A. Vanelli-Coralli, Adaptive coding and modulation techniques for next generation hand-held mobile satellite communications, IEEE International Conference on Communications (ICC), 9 June 2013. doi:10.1109/ICC.2013.6655277.
[11] S. Salih, M.M.A. Suliman, Implementation of adaptive modulation and coding techniques using Matlab, Proceedings ELMAR-2011, pp. 137-139, 2011.
[12] L. Srividya, P.N. Sudha, Adaptive Channel Coding to Enhance the Performance in Rayleigh Channel, Int. J. of Adv. Comput. Sci. and Appl., vol. 15, no. 6, pp. 504–511, 2024. doi: 10.14569/IJACSA.2024.0150652.
[13] S. Kumari, L. Gahalod, S.Changlani, Study of Different Types of Error Detection and Correction Code in Wireless Communication, Int. J. of Scientific Res. in Sci., Eng. and Tech., pp. 448-455, 2022. doi: 10.32628/IJSRSET2293138.
[14] P. Santini, M. Battaglioni, M. Baldi, and F.Chiaraluce, Analysis of the error correctioncapability of LDPC and MDPC codes under parallelbit-flipping decoding and application tocryptography, IEEE Trans. Communication, vol.68, no. 8, pp. 4648_4660, 2020.
[15] K. Pearson, On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, Philosophical Magazine, Series 5, vol. 50, no. 302, pp. 157–175, 1900.
[16] M.Z. Yakubova, O.A. Manankova, K.A. Tashev, K.A., G.S.Sadikova, Methodology of the determining for Pearson's criterion based on researching the value of delays in the transmitting of information over a multiservice network, 2020 International Conference on Information Science and Communications Technologies, ICISCT 2020, 2020. doi: 10.1109/ICISCT50599.2020.9351419.
[17] B. Lee, S. Park, D. J. Love, H. Ji, and B. Shim, Packet Structure and Receiver Design for Low Latency Wireless Communications With Ultra-Short Packets, in IEEE Transactions on Communications, vol. 66, no. 2, pp. 796-807, Feb. 2018, doi: 10.1109/TCOMM.2017.2755012.
[18] E. Faure, A. Shcherba, B. Stupka, Permutation-based frame synchronisation method for short packet communication systems, In: Proc. 11th IEEE Intl. Conf. on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Cracow, Poland, 22-25 Sept., p. 1073–1077.
[19] E. Faure, V. Shvydkyi, A. Shcherba, O. Kharin, B. Stupka, Method of cyclic synchronization based on permutations, Bull. Cherkasy State Technol. Uni., vol. 4, November 2020, pp. 67–76. doi: 10.24025/2306-4412.4.2020.222439.
[20] J. Al-Azzeh, E. Faure, A. Shcherba and B. Stupka, Permutation-based frame synchronization method for data transmission systems with short packets, Egypt. Inform. J., vol. 23, no. 3, September 2022, pp. 529–545. doi: 10.1016/j.eij.2022.05.005.
[21] E. Faure, A. Shcherba, Y. Vasiliu, and A. Fesenko, Cryptographic key exchange method for data factorial coding, in CEUR Workshop Proceedings, vol. 2654, pp. 643-653, 2020.
[22] E. Faure, A. Shcherba, A. Kharin, Factorial code with a given number of inversions, Radio Electron. Comput. Sci. Control, vol. 2, no. 2, pp. 143–53, 2018.
[23] E. Faure, V. Shvydkyi, A. Lavdanskyi, O. Kharin, Methods of factorial coding of speech signals, Radio Electron. Comput. Sci. Control, vol.4, no. 4, pp.186–98, 2019.
[24] J.S. Al-Azzeh, B. Ayyoub, E. Faure, V. Shvydkyi, O. Kharin, A. Lavdanskyi, Telecommunication Systems with Multiple Access Based on Data Factorial Coding, in International Journal on Communications Antenna and Propagation (IRECAP), vol. 10, no. 2, pp. 102-113, 2020, doi: 10.15866/irecap.v10i2.17216.
[25] G. Durisi, T. Koch, and P. Popovski, Toward Massive, Ultrareliable, and Low-Latency Wireless Communication with Short Packets, Proceedings of the IEEE, vol. 104, no. 9, pp. 1711-1726, 2016.
[26] K.-U. Schmidt, Low-correlation sequences, Des. Codes Cryptog., vol. 78, no. 1, pp. 237–267, 2016. doi: 10.1007/s10623-015-0154-7.