http://doi.org/10.35784/iapgos.5762

received: 05.01.2024 | revised: 04.04.2024 | accepted: 09.04.2024 | available online: 30.06.2024

ITERATIVE DECODING OF SHORT LOW-DENSITY PARITY-CHECK CODES BASED ON DIFFERENTIAL EVOLUTION

Mykola Shtompel, Sergii Prykhodko

Ukrainian State University of Railway Transport, Department of Transport Communication, Kharkiv, Ukraine

Abstract. *To ensure a given quality of service in the networks of the Internet of Things, short error-correcting codes are used, in particular, low-density parity-check codes. The paper proposes an approach for decoding these codes based on the joint application of belief propagation and differential evolution procedures. It is shown that in order to reduce the search area of error vectors based on differential evolution, it is necessary to use the least reliable basis of the parity-check matrix. Flowchart and pseudocode of the combined decoding algorithm of short low-density parity-check codes were presented. The simulation results showed that the proposed decoding method provides an additional gain from encoding compared to the classical decoding method. The application of the presented iterative decoding method of short low-density parity-check codes will improve the efficiency of data transmission in the infrastructure of the Internet of Things.*

Keywords: Internet of Things, low-density parity-check codes, iterative decoding, differential evolution

ITERACYJNE DEKODOWANIE KRÓTKICH KODÓW PARZYSTOŚCI O NISKIEJ GĘSTOŚCI W OPARCIU O EWOLUCJĘ RÓŻNICOWĄ

Streszczenie. *Aby zapewnić określoną jakość usług w sieciach Internetu Rzeczy, stosowane są krótkie kody korekcji błędów, w szczególności kody kontroli parzystości o niskiej gęstości. W artykule zaproponowano podejście do dekodowania tych kodów oparte na wspólnym zastosowaniu procedur propagacji zaufania i ewolucji różnicowej. Pokazano, że w celu zmniejszenia obszaru wyszukiwania wektorów błędów w oparciu o ewolucję różnicową, konieczne jest użycie najmniej wiarygodnej podstawy macierzy kontroli parzystości. Przedstawiono schemat blokowy i pseudokod połączonego algorytmu dekodowania krótkich kodów kontroli parzystości o niskiej gęstości. Wyniki symulacji wykazały, że proponowana metoda dekodowania zapewnia dodatkowy zysk z kodowania w porównaniu z klasyczną metodą dekodowania. Zastosowanie przedstawionej iteracyjnej metody dekodowania krótkich kodów o niskiej gęstości parzystości poprawi wydajność transmisji danych w infrastrukturze Internetu Rzeczy.*

Słowa kluczowe: Internet rzeczy, kody o niskiej gęstości parzystości, dekodowanie iteracyjne, ewolucja różnicowa

Introduction

The Internet of Things is an important concept for machine-to-machine communication based on wireless telecommunication technologies. Data transfer between equipment of the Internet of Things is carried out in small packets over low-speed communication channels. In addition, equipment often has limited computing resources and increased power consumption requirements [3]. The presented features of the technical implementation of the infrastructure of the Internet of Things complicate the provision of a given quality of service, in particular, the reliability of data transmission.

To solve this problem in the field of the Internet of Things, various short error-correcting codes are used, the length of which is tens to hundreds of bits. Promising codes with good performance are low-density parity-check (LDPC) codes. For the mathematical description of these codes, matrix or graph approaches are used. According to the matrix approach, the parameters of certain LDPC code are determined by a sparse parity-check matrix or an equivalent generator matrix. The graph approach is based on the representation of the LDPC codes using the Tanner graph, which consists of bit and check nodes. The classic method for decoding these codes is a iterative belief propagation decoding. For long codes, this approach ensures the achievement of the Shannon bound for different communication channel models. However, for short codes, there is a significant degradation in correction capability when using this decoding method due to the presence of cycles in the corresponding Tanner graph [5].

Therefore, a promising direction for ensuring the given reliability and increasing efficiency of data transmission in the networks of the Internet of Things using short LDPC codes is the search for a more efficient decoding method.

1. Problem definition

To increase the efficiency of decoding short LDPC codes, a combination of iterative belief propagation decoding with additional computational and optimization procedures is used.

Recently, neural networks have been widely used in the field of decoding LDPC codes.

In [1], an improved approach to the decoding of short LDPC codes based on belief neural propagation is proposed. At the first stage of decoding, the least reliable element of the codeword is determined, and at the second stage, a decision is made by the neural network on the decimation of some element of the codeword.

In [6], a method for decoding non-binary LDPC codes using neural networks is presented. It is shown that the use of a genetic algorithm makes it is possible to increase the efficiency of training the weights of neural networks and provide a lower decoding error probability.

It is possible to increase the efficiency of decoding short LDPC codes by parallel using of several decoders built on the basis of belief propagation and recurrent neural networks. In the basic version of this approach, decoders are trained for different types of errors [9]. A further development of this approach is to create decoder diversity architectures together with a procedure based on ordered statistics, which makes it is possible to approach the maximum likelihood decoding performance [10].

The presented neural network decoding methods require complex learning methods, significant computational resources, and the need to adapt to the code structure, so other approaches have been proposed.

For example, the application of the ordered statistics procedure and partial cyclic redundancy coding together with the classical belief propagation decoding can significantly increase the efficiency of short LDPC codes [4].

In [2], an efficient approach for decoding short LDPC codes based on parallel genetic algorithms with the use of special crossover and selection operators is presented.

Another example of the applying of bioinspired procedures is the use of differential evolution in decoding a special class of error-correcting codes – algebraic convolutional codes [7]. The results obtained in the paper show that this approach can be used to increase the efficiency of decoding these codes in comparison with the algebraic decoding method.

Therefore, taking into account the presented results, in order to further improve the efficiency of using short LDPC codes in the field of the Internet of Things, it is proposed to combine belief propagation decoding with differential evolution procedure. In this paper, the additional energy gain from encoding provided by developed decoding method of LDPC codes is defined as a criterion of data transmission efficiency.

2. Proposed decoding method of LDPC codes

Let the binary (N,K) LDPC code be given by a parity-check matrix H of size $M \times N$, where N is the codeword length, K is the message length, $M=(N-K)$ is the number of check symbols. Also assume that binary phase modulation is used, so the binary codeword $x=(x_1,x_2,...,x_N)$ is mapped to the bipolar codeword $\bar{b}=(b_1,b_2,...,b_N)$ based on the transformation $b_i = 1-2x_i$, $i \in [1,N]$. After transmitting this codeword through a communication channel with additive white Gaussian noise (AWGN), we get the received word $y=(y_1,y_2,...,y_N)$, where $y_i = b_i + z_i$ is an independent random Gaussian value with zero mean and variance $\mathrm{N}(0,\sigma^2)$, $i \in [1,N]$.

The main steps of the proposed decoding method of short LDPC codes are presented below.

Stage 1. Decoding based on belief propagation.

The parity-check matrix H corresponds to the Tanner graph consisting of V bit and C check nodes. Let $L^C_{j\to i}$ is the message from the c_j check node to the v_i bit node, $L^V_{i\to j}$ is the message from the v_i bit node to the c_j check node, L^t_i is the decoder decision for the v_i bit node on the t iteration. Let's also denote $M(i)$ is the set of indices of check nodes connected to the v_i bit node, $N(j)$ is the set of indices of bit nodes connected to the c_j check node. We define $M(i)\setminus j$ and $N(j)\setminus i$ as the set $M(i)$ excluding j and the set $N(j)$ excluding i, respectively.

First, the decoder initializes the initial value of the log-likelihood ratio for the received word:

$$L^V_{i\to j} = \ln\left(\frac{P(b_i=1\,|\,y_i)}{P(b_i=-1\,|\,y_i)}\right) = \frac{2y_i}{\sigma^2} \tag{1}$$

Further, at each decoding iteration t, messages are exchanged between c_j check and v_i bit nodes of the Tanner graph, which are calculated as follows:

$$L^C_{j\to i} = 2\tanh^{-1}\left(\prod_{i'\in N(j)\setminus i}\tanh\left(\frac{1}{2}L^V_{j'\to i}\right)\right) \tag{2}$$

$$L^V_{i\to j} = \frac{2y_i}{\sigma^2} + \sum_{j'\in M(i)\setminus j} L^C_{j'\to i} \tag{3}$$

After reaching the maximum number of iterations T, based on expressions (2) and (3), taking into account (1), the final value of the log-likelihood ratio for the received word is formed:

$$L_i = \frac{2y_i}{\sigma^2} + \sum_{j\in M(i)} L^C_{j\to i} \tag{4}$$

Further, based on (4), a hard decision is made and an estimate of the codeword is formed:

$$\hat{x}_i = \mathrm{sign}(L_i) = \begin{cases} 0, & L_i \ge 0 \\ 1, & L_i < 0 \end{cases} \tag{5}$$

In addition, at this stage, the decoder calculates the reliability of the values for each bit node v_i:

$$r_i = \left|\sum_{t=0}^{T} \alpha^{T-t} L^t_i\right| \tag{6}$$

where α is the attenuation coefficient, the value of which depends on the characteristics of the code and the number of decoding iterations [4].

If the obtained estimate of the codeword (5) satisfies the condition $\hat{x}H^T = 0$, then the decoding is completed, otherwise, the total reliability value (6) for the bit nodes is saved and the transition to the next stage is performed.

Stage 2. Finding the least reliable basis of the parity-check matrix.

Suppose that the parity-check matrix H has full rank, then the formation of the least reliable basis is carried out by applying such permutations [11]. The first permutation π_1 corresponds to sorting the elements $\hat{e}_i$, $i \in [1,N]$ of the error vector in order of increasing reliability of the values (6). By applying this permutation to the columns of the original matrix H, we obtain the matrix $H^{(1)}$. Further, using the permutation π_2, the matrix $H^{(2)}$ is formed, in which the M left columns are independent, and the remaining columns form an information set. Thus, the trial error vector is equal to $\hat{e}^{(2)} = \pi_2(\pi_1(\hat{e}))$.

To convert the matrix $H^{(2)}$ into a systematic form, the Gaussian elimination method is used, taking into account the codeword estimate (5) obtained at Stage 1:

$$\begin{aligned} &E\hat{x} = EH^{(2)}\hat{e}^{(2)} \Rightarrow \\ &E\hat{x} = [IEH^{(2)}_{IS}][\hat{e}^{(2)}_I \hat{e}^{(2)}_{IS}]' \Rightarrow \\ &\hat{e}^{(2)}_I = EH^{(2)}_{IS}\hat{e}^{(2)}_{IS} + E\hat{x} \end{aligned} \tag{7}$$

where E is the equivalent matrix for implementing Gaussian elimination; I is the identity matrix of size $M \times M$; $H^{(2)}_{IS}$ is the information set matrix of size $(N-K)\times K$; $\hat{e}^{(2)}_I$, $\hat{e}^{(2)}_{IS}$ are the components of the trial error vector $\hat{e}^{(2)}$.

Stage 3. Search for an estimate of the trial error vector based on differential evolution.

In the used version of the differential evolution procedure, the initial population is formed randomly from N_p vectors $\hat{e}^{(2)}_{IS}$, including the zero vector, which corresponds to the case of no errors. After that, based on (7), possible trial error vectors $\hat{e}^{(2)}$ are found.

To assess the quality of the obtained error vectors, a fitness function based on the mismatch rule is used:

$$D(\bar{y},\hat{e}) = \min \sum_{i:\hat{e}_i=1} |\,y_i\,| \tag{8}$$

where $\hat{e} = \pi_1^{-1}(\pi_2^{-1}(\hat{e}^{(2)}))$.

Futher, the next population is formed by applying to the current vectors $\hat{e}^{(2)}_{IS}$ the mutation, crossover, and selection operators, the implementation features of which are presented in [8]. According to differential evolution procedure for each trial vector $\hat{e}^{(2)}_{IS}$ in population N_p the new value is calculated based on several random vectors and specific tuning parameters. These parameters are crossover probability CR and differential weight F that should be selected properly to improve quality of error vectors searching. This process is repeated iteratively, and when the maximum number of iterations L is reached, the most probable trial error vector $\tilde{e}^{(2)}$ with the best quality based on (8) is determined.

Stage 4. Formation of the estimate of the transmitted codeword.

The estimate of the transmitted codeword is found as $\tilde{x} = \hat{x} + \tilde{e}$, where $\hat{x}$ is the hard solution (5) obtained at Step 1; $\tilde{e} = \pi_1^{-1}(\pi_2^{-1}(\tilde{e}^{(2)}))$ is the error vector, which is determined by the inverse transformation of the trial error vector found in Stage 3.

Thus, the proposed decoding method of short LDPC codes involves a combination of belief propagation and differential evolution procedures. First, a specified number of iterations of belief propagation decoding is performed and the reliability of the estimated codeword is defined. Further, the search for the most probable error vector for the found codeword is carried out. For this, the least reliable basis of the parity-check matrix based on two permutations is determined and the differential evolution procedure is applied iteratively to evaluate the quality of the different trial error vectors. After reaching the maximum number of iterations of differential evolution the most probable error vector is found. Finally, the transmitted codeword is defined by apllying the transformed error vector to estimated codeword found by belief propagation decoding.

3. Experiments and results

The computer model of the wireless network system of the Internet of Things was created to evaluate the energy effectiveness of the developed decoding method of LDPC codes. This model uses binary phase modulation to transmit signals through AWGN communication channel with specified characteristics. To investigate features and limitations of proposed approach, as well as to calculate the additional energy gain from encoding, the possibility of changing code parameters and decoding parameters is assumed.

In order to conduct experimental research using this model, a software implementation of presented decoding method of short LDPC codes was developed.

The flowchart of the proposed decoding algorithm of LDPC codes is presented in Fig. 1.

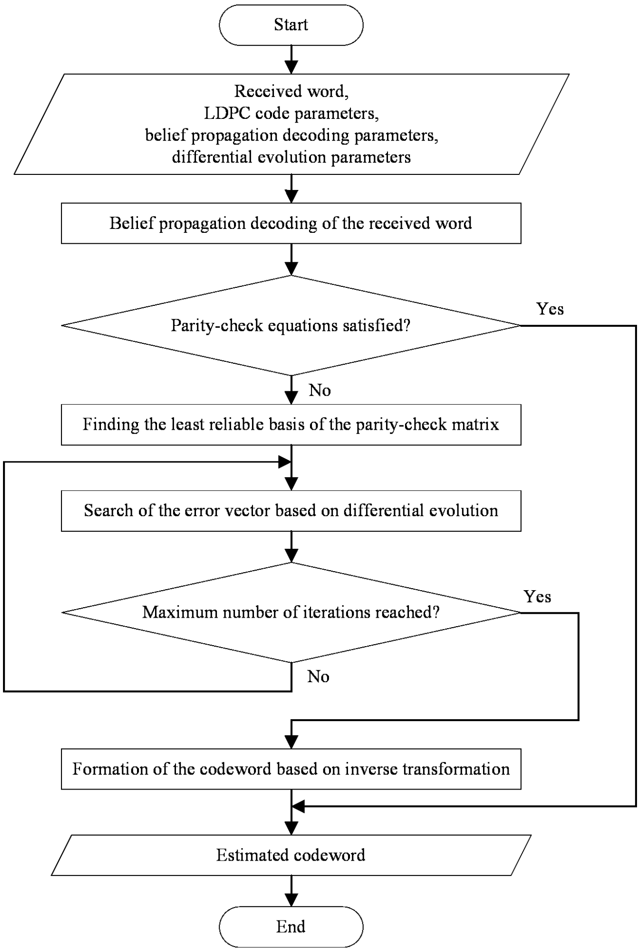

Fig. 1. Flowchart of the combined decoding algorithm of short LDPC codes

From Fig. 1 it follows that the received word from the communication channel, the parameters of used LDPC code, belief propagation and differential evolution procedures are input data for this algorithm. First, the sub-process of conventional belief propagation decoding is performed. As a result of this sub-process, the log-likelihood ratio vector, the reliability vector, and the corresponding binary codeword are formed. If the parity-check equations are not equal to zero, then the transition to the sub-process of decoding based on differential evolution search is made. This sub-process uses the formed reliability vector as additional input information. After reaching the maximum number of iterations of differential evolution, the most probable error vector based on the modified parity-check matrix is formed. As a result of the implementation of this sub-process, the codeword is formed by applying the inverse transformation to the found error vector. The output of this decoding algorithm is an estimate of the most probable codeword, which is chosen as the transmitted codeword.

The pseudocode of this decoding algorithm is presented in Fig. 2.

Input: received word $y = (y_1, y_2, ..., y_N)$, LDPC code parameters N, K, H, decoding parameters T, α, N_p, L, F, CR

Output: estimated codeword $\tilde{x}$

Initialize values for the received word $L^V_{i\to j}$ as in (1)
for each iteration $t \in [1, T]$ **do**
  Calculate values for check and bit nodes of the Tanner graph $L^C_{j\to i}$ and $L^V_{i\to j}$ based on (2) and (3)
  Calculate the reliability of the values for each bit node r_i based on (6)
end for
Form final values for the received word L_i based on (4)
Find hard decision of estimated codeword $\tilde{x}$ as in (5)
if $\hat{x}H^T = 0$ **then**
  Return estimated codeword $\tilde{x}$
else
  Save total reliability values r_i for the bit nodes as in (6)
  Form the error vector $\hat{e}$ in order of increasing reliability by applying permutation π_1 and obtaining the matrix $H^{(1)}$
  Form the trial error vector $\hat{e}^{(2)}$ by applying permutation π_2 and obtaining the matrix $H^{(2)}$
  for each iteration $l \in [1, L]$ **do**
    Generate N_p random vectors $\hat{e}^{(2)}_{IS}$ (including the zero vector) and form trial error vectors $\hat{e}^{(2)}$ based on (7)
    Apply differential evolution operators with parameters F and CR to the current error vectors $\hat{e}^{(2)}$
    Calculate the fitness function $D(y, \hat{e})$ based on (8) to asset the quality of the obtained error vectors
  end for
  Determine the best error vector $\tilde{e}$ by the inverse transformation
  Find estimated codeword as $\tilde{x} = \hat{x} + \tilde{e}$
end if

Fig. 2. Pseudocode of the combined decoding algorithm of short LDPC codes

The simulation was carried out for the AWGN communication channel for binary (64, 32) and (128, 64) short LDPC codes.

When implementing the developed decoding method, the following parameters were used: the maximum number of belief propagation decoding iterations $T = 200$; attenuation coefficient $\alpha = 1$; differential evolution parameters – population size $N_p = 40$, maximum number of iterations $L = 100$, differential weight $F = 0.7$, crossover probability $CR = 0.8$. To ensure the reliability of the results obtained, at least 100 decoding errors were recorded for each value of the signal-to-noise ratio.

Simulation results of the proposed decoding method (belief propagation-differential evolution, BP-DE), standard decoding based on the belief propagation (BP), and maximum likelihood (ML) decoding for (64, 32) and (128, 64) LDPC codes are shown in Fig. 3 and Fig. 4, respectively.

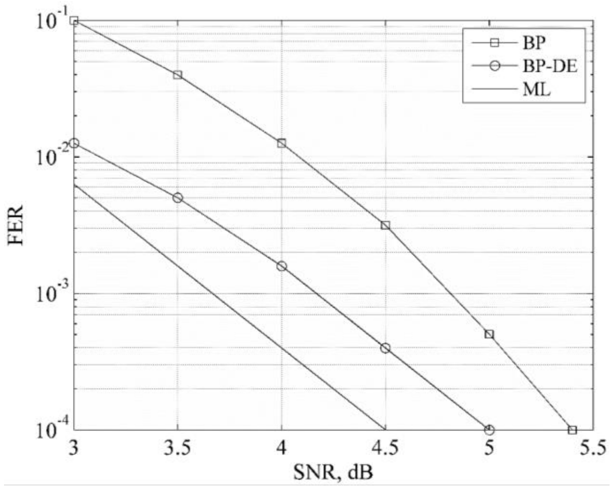

Fig. 3. Dependence of frame error rate (FER) on signal-to-noise ratio (SNR) for (64, 32) LDPC code

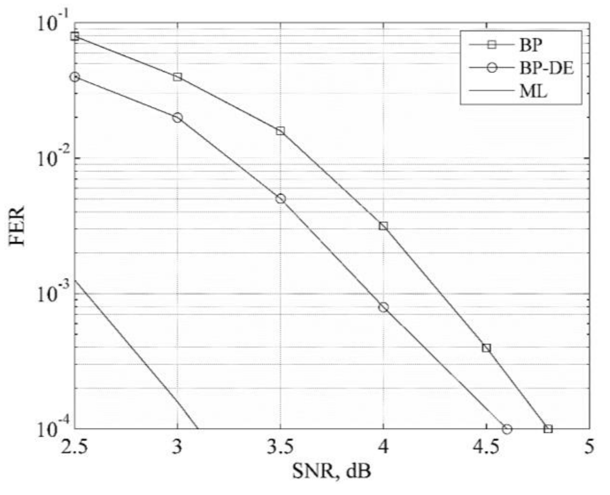

Fig. 4. Dependence of the frame error rate (FER) on the signal-to-noise ratio (SNR) for (128, 64) LDPC code

It follows from Fig. 3 and Fig. 4 that the proposed decoding method provides higher energy efficiency, which is measured in dB, compared to the classical decoding method based on belief propagation. For example, for the (64, 32) LDPC code, the energy gain from encoding with the frame error rate $FER = 10^{-4}$ is 0.4 dB, and for the (128, 64) LDPC code is 0.2 dB. On the other hand, the obtained results show that the energy efficiency of the proposed decoding method decreases for longer LDPC codes. In particular, for (128, 64) LDPC code with frame error rate $FER = 10^{-4}$, the difference in the required signal-to-noise ratio between the developed decoding method and maximum likelihood decoding is about 1.5 dB.

4. Conclusion

Short LDPC codes are used to increase the reliability of data transmission and improve the energy performance of equipment in wireless networks of the Internet of Things. However, the efficiency of the classical belief propagation decoding method is significantly limited for short codes.

The paper proposes an approach to decoding short LDPC codes based on the joint application of belief propagation and differential evolution procedures. These codes have a sparse parity-check matrix, therefore, to reduce the search area for error vec-tors based on differential evolution the least reliable basis is constructed. The simulation results in the AWGN communication channel showed that the proposed decoding method provides an additional energy gain from encoding compared to the classical decoding method. However, with an increasing the code length, the performance of the developed decoding method decreases, which does not allow reaching the maximum likelihood decoding limit.

Thus, the presented iterative decoding method using the differential evolution procedure makes it is possible to increase the efficiency of data transmission in the infrastructure of the Internet of Things.

References

[1] Buchberger A. et al.: Learned decimation for neural belief propagation decoders: Invited paper. ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, 8273–8277.
[2] Chaibi H. et al.: Parallel genetic algorithm decoder scheme based on DP-LDPC codes for industrial IoT scenarios. Procedia Computer Science 176, 2020, 3496–3505.
[3] Chettri L., Bera R.: A comprehensive survey on internet of things (IoT) toward 5G wireless systems. IEEE Internet of Things Journal 7(1), 2019, 16–32.
[4] Jiang M. et al.: Partial CRC-aided decoding of 5G-NR short codes using reliability information. Science China Information Sciences 62, 80303, 2019.
[5] Khan B. S. et al.: URLLC and eMBB in 5G industrial IoT: A survey. IEEE Open Journal of the Communications Society 3, 2022, 1134–1163.
[6] Liu T., Chen X.: Deep learning-based belief propagation algorithm over non-binary finite fields. International Conference on Wireless Communications and Signal Processing (WCSP), 2020, 164–169.
[7] Panchenko S. et al.: Analysis of efficiency of the bioinspired method for decoding algebraic convolutional codes. Eastern-European Journal of Enterprise Technologies 2-4(98), 2019, 22–30.
[8] Price K., Storn R. M., Lampinen J. A.: Differential evolution: A practical approach to global optimization. Springer 2005.
[9] Rosseel J. et al.: Error structure aware parallel BP-RNN decoders for short LDPC codes. 2021 11th International Symposium on Topics in Coding (ISTC), 2021, 1–5.
[10] Rosseel J. et al.: Decoding short LDPC codes via BP-RNN diversity and reliability-based post-processing. IEEE Transactions on Communications 70(12), 2022, 7830–7842.
[11] Watanabe K. et al.: Shortened LDPC codes accelerate OSD decoding performance. EURASIP Journal on Wireless Communications and Networking 22, 2021.

D.Sc. Mykola Shtompel
e-mail: shtompel.mykola@kart.edu.ua

Doctor of Technical Sciences, Professor of Transport Communication Department, Ukrainian State University of Railway Transport, Ukraine.
The research interests are related to the field of telecommunication and information technologies, signal processing and error-correcting codes.

http://orcid.org/0000-0003-3132-8335

D.Sc. Sergii Prykhodko
e-mail: prihodko@kart.edu.ua

Doctor of Technical Sciences, Head of Transport Communication Department, Ukrainian State University of Railway Transport, Ukraine.
The research interests are related to the field of mathematical and computer modelling of telecommunication systems and networks.

http://orcid.org/0000-0001-6535-8351