Some Comments About CRC Selection for the 5G NR Specification

TSONKA BAICHEVA, PETER KAZAKOV, AND MIROSLAV DIMITROV

Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, 1113 Sofia, Bulgaria

Corresponding author: Miroslav Dimitrov (mirdim@math.bas.bg)

This work was supported by the National Science Program Security and Defense (NSP SD) Program, funded by the Ministry of Education and Science of the Republic of Bulgaria under Grant D01-74/19.05.2022.

ABSTRACT

In the 5G new radio technical specification, 6 cyclic redundancy-check (CRC) codes of 6, 11, 16, and 24 check bits are proposed to be used for error detection. In this work, we consider all CRC codes with the proposed number of check bits and determine those having the best error detection performance in the whole interval of lengths for which, according to the specification, they are supposed to work. To evaluate the error detection performance of the investigated codes we calculate the cumulative value $S_d$ of the number of codewords of minimum weight for the whole interval of lengths they will work. We choose as the best code the one having the smallest $S_d$. The determined best CRC codes of 6, 11, and 16 check bits have better error detection performance than those proposed in the specification. For 24 check bits, 3 CRC codes are presented, but only one the best for the whole interval of lengths code was not found.

INDEX TERMS

5G NR specification, binary symmetric channel, CRC codes, undetected error probability.

I. INTRODUCTION

According to the technical specification of the 5G new radio (5G NR) [1], data and control streams from/to the MAC layer are encoded/decoded to offer transport and control services over the radio transmission link. Channel coding scheme combines error detection, error correction, rate matching, interleaving, and transport channel or control information mapping onto/splitting from physical channels. For error detection cyclic redundancy-check (CRC) codes or a combination of LDPC and CRC codes are used. Six CRC codes are proposed in the specification and are specified to work for different payloads (information stream lengths) $A$.

The proposed CRC codes are summarized in Table 1.

| Label   | Polynomial          |
|---------|---------------------|
| CRC6    | $x^6 + x^5 + 1$    |
| CRC11   | $x^{11} + x^{10} + x^9 + x^8 + 1$ |
| CRC16   | $x^{16} + x^{12} + x^8 + 1$ |
| CRC24A  | $x^{24} + x^{23} + x^{18} + x^{17} + x^{14} + x^{11} + x^{10} + x^7 + x^6 + x^5 + x^4 + x^3 + x + 1$ |
| CRC24B  | $x^{24} + x^{23} + x^6 + x^5 + x + 1$ |
| CRC24C  | $x^{65} + x^{64} + x^{61} + x^{60} + x^{57} + x^{56} + + x^{13} + x^{12} + x^9 + x^4 + x^3 + x + 1$ |

TABLE 1. CRC codes used in 5G NR.

For the uplink and downlink shared channels, CRC16 should be used if the transport block size $A$ is $1 \leq A \leq 3824$, while for $3824 \leq A \leq 8424$, CRC24A is proposed. For the downlink broadcast channel, where $A \leq 8424$, CRC24C is suggested. Error detection in the uplink control channel for $12 \leq A \leq 19$ should be done by CRC6, for $20 \leq A \leq 1706$ by CRC11 while in the downlink control channel, where $A \leq 140$, CRC24C should be used.

In this paper, we investigate the error detection performance of the aforementioned CRC codes. For every number of check bits, i.e. 6, 11, 16, and 24, we consider all CRC codes that can be used for the proposed interval of lengths. For each code, we calculate the sum of its number of codewords of minimum weight for all lengths from the interval. The code having the smallest sum will have the best error detection...
performance for the considered intervals of lengths. This way we determine the best CRC code for the given number of check bits.

The work is organized as follows. First, we briefly recall the basic facts about the generation and error detection performance of CRC codes. We consider the complexity of the determination of the undetected error probability, which is a speed-up version of the one from [2]. Since the CRC codes are proposed to be used at any of the lengths specified in 5G NR, we investigate their cumulative performance for the whole interval of lengths and use it as an optimization criterion. As a result, we suggest a collection of six CRC codes that outperform, in terms of error detection capabilities, those proposed in 5G NR. Although in [2] we obtained the best CRC codes for 11 and 16 check bits, we did the investigation again as in the 5G NR specification they have to work in a different range of lengths. It can be seen from the results that we obtained another best CRC code with 11 check bits.

II. BASIC FACTS ABOUT CRC CODES

CRC codes are first introduced by Peterson and Brown in [3]. Very good sources describing the theoretical foundations and properties of CRC codes are Peterson and Weldon’s [4] and Wicker’s [5] books. In this section, the main results regarding the structure and error detection properties of CRC codes are summarized and introduced.

CRC codes are full-length or shortened linear binary cyclic codes. Each codeword of a CRC code has \( n = k + p \) binary digits and is obtained by adding in a definite way a block of \( p \) parity bits to an information block of \( k \) binary digits.

Any CRC code could be represented as a set of polynomials where instead of the information block of \( k \) binary digits, the polynomial

\[
i(x) = i_0 + i_1 x + \cdots + i_{k-1} x^{k-1},
\]

and instead of the block of \( p \) parity bits the polynomial

\[
r(x) = r_0 + r_1 x + \cdots + r_{p-1} x^{p-1}
\]

are used. Then, in each CRC code with \( p \) check bits, there exists a polynomial of degree \( p \) called the generator polynomial \( g(x) \) of the code.

Each codeword \( c(x) \) of the CRC code is obtained by computing \( r(x) \) from \( i(x) \) in the following way

\[
r(x) \equiv (x^p \cdot i(x)) \ mod \ g(x).
\]

According to this encoding the following equations holds

\[
c(x) = x^p i(x) + r(x) = q(x) g(x) + r(x) + r(x) = q(x) g(x)
\]

for some polynomial \( q(x) \). Thus, the recipient could easily decide if a given word is a codeword or not by checking its divisibility to \( g(x) \).

The polynomial \( g(x) \) divides \( x^{n_c} + 1 \), where

\[
n_c = \min\{m | x^m \equiv 1 \ mod \ g(x)\}
\]
is called order of the polynomial \( g(x) \). The binary cyclic code \( D \) generated by the polynomial \( g(x) \) of degree \( p \) is a \([n_c, n_c-p] \) code. Any \([n, n-p] \) subcode \( C \) of \( D \) obtained by shortening \( D \) in \( n_c - n \) positions is a CRC code. The maximum length at which any CRC code can be used is equal to the order of the generator polynomial \( n_c \). For longer lengths, the code is no more a CRC code. More precisely, it becomes a repetition code with minimum distance 2. This means that it can detect only 1 error and does not possess the ability to detect burst errors.

Given their excellent error detection capabilities, CRC codes are widely exploited by manifold applications.

Let us denote by \( d \) the minimum distance of an \([n, n-p] \) CRC code generated by the polynomial \( g(x) \) and by \( A_i \) the number of codewords of weight \( i \) in the code. The error detection performance of a CRC code is measured by the number of errors it is not able to detect, i.e. by its undetected error probability \( P_{ue} \). Thus we are looking for code that will be not able to detect as small as possible number of errors. CRC codes are linear codes and therefore they will be not able to detect only error patterns which are codewords. If we denote by \( \varepsilon \) the symbol error probability of the communication channel, then we can express the probability of undetected errors of a CRC code as

\[
P_{ud} = \sum_{i=d}^{n} A_i \varepsilon^i (1-\varepsilon)^{n-i}.
\]

III. COMPLEXITY OF THE DETERMINATION OF \( P_{UE} \) OF A CODE

In [6], it has been proven that the determination of the set \( \{A_i\}_{i=1}^{n} \) for a linear \([n, n-p] \) code is a computationally hard problem. Thus the comparison of the values of \( P_{ue} \) for two particular codes is also a computationally hard problem.

The hardness of the determination of \( P_{ue} \) of a CRC code is investigated in many works. Fujiwara et al. [7] have developed a method for evaluation of \( P_{ue} \) for shortened Hamming codes which are CRC codes with primitive generator polynomials. A similar method for even weights shortened Hamming codes is suggested by Wolf and Blakeney [8]. Castagnoly et al. [9] extend this method for arbitrary CRC codes. Chun and Wolf [10] describe a hardware device for efficient evaluation of \( P_{ue} \) for a class of CRC codes with a large number of parity check digits. The generator polynomials for the codes in this class are of the form \( g(x) = (1+x)p(x) \) where \( p(x) \) is a primitive irreducible polynomial.

In this work we use the approach from [11] to reduce the computational complexity of the calculation of \( P_{ue} \). Namely, instead of \( P_{ue} \) we compare \( P_{ue}' = A_d \varepsilon^d (1-\varepsilon)^{n-d} \), i.e. only the first nonzero addend of the sum for \( P_{ue} \), as it is the most significant one (especially for small values of \( \varepsilon \)). This approach allows us to significantly decrease the computational time as we need to determine only the number of codewords with minimum weight.
As it can be seen in Figure 1, comparing $P_{ue}$ performances of two codes we can obtain rather precise results, despite that the actual difference between the the undetected error probabilities of the codes is several times bigger. This approach has been used by Kazakov in [11] to determine optimal with respect to $A_d$ CRC codes with 16 check bits.

![Comparison of $P_{ue}$ and $P_{ue}'$ of two CRC codes.](image)

**Figure 1.** Comparison of $P_{ue}$ and $P_{ue}'$ of two CRC codes.

### IV. SEARCH FOR OPTIMAL CRC CODES WITH $P$ CHECK SYMBOLS

The proposed in the 5G NR specification CRC codes will be used to protect information packages of different lengths. Having this in mind, we aim to choose a CRC code that has the best performance not only for a particular length but for the whole range of lengths at which the code is supposed to work.

We denote by $d(C_{g,n})$ the minimum distance of a CRC code of length $n$, with generator polynomial $g(x)$, supposed to work in an interval of lengths between $L$ and $M$. Since the CRC code can be used at any length of the interval $[L \ldots M]$ its cumulative performance in the whole interval should be investigated. Thus, for any polynomial $g(x)$ of degree $p$, which could be the generator polynomial of a CRC code of length $n \in [L \ldots M]$ we calculate

$$S_d = \sum_{n=L}^{M} d(C_{g,n}).$$

The CRC code with a maximum value of $S_d$ is the optimal one.

To obtain $d(C_{g,n})$, we have to generate all $2^{n-p}$ codewords of the CRC code and determine their weights as the minimum distance of a binary linear code is equal to the minimum weight of its codewords. Given some fixed CRC code, to calculate $S_d$ we have to repeat this procedure $M - L + 1$ times. However, instead of generating the $2^{n-p}$ codewords of the CRC code $C$, we can generate only the $2^p$ codewords of its dual code $C'$ and compute the minimum distance of $C$ in linear time via MacWilliams’ identities [12].

To calculate the values of $S_d$ for all CRC codes with $p = 6, 11, 16, 24$ parity bits, we apply the same divide and conquer technique suggested in [2]. To speed up additionally the algorithm, we use new early rejection criteria by requiring a minimum distance 6 for large code lengths and further apply some improved programming techniques. Then, for each $p = 6, 11, 16, 24$, we determine the CRC code having maximum $S_d$. If we obtain more than one code of maximum $S_d$, we consider those having the smallest sum of the numbers of codewords with minimum weight, i.e. minimum $S_{A_d} = \sum_{n=L}^{M} A_d$.

### V. RESULTS

In Table 2, we summarize the range of code lengths at which the proposed in 5G NR technical specification CRC codes are defined to be used.

**Table 2.** Lengths at which CRC codes from 5G NR specification are applicable.

| CRC code | $L$ | $M$ |
|----------|-----|-----|
| CRC6     | 18  | 25  |
| CRC11    | 31  | 1717|
| CRC16    | 17  | 3840|
| CRC24A   | 3848| 8448|
| CRC24B   | 25  | 8448|
| CRC24C   | 25  | 164 or 8448|

For CRC codes with $p = 6, 11, 16$ or 24 parity bits, we determine the best ones, by following the strategy described throughout the previous section. We consider only CRC codes of lengths in the interval $[L \ldots M]$ according to Table 2. Given a fixed number of parity bits $p$, we have to consider $2^{p-1}$ polynomials - possible generators of a CRC code. However, we could reduce this number by omitting the reciprocal ones, since they generate equivalent codes as well as codes with orders smaller than $M$.

#### A. CRC CODES OF $P = 6$

All polynomials of degree 6 that could generate CRC codes are $2^5 = 32$. However, after excluding the reciprocals they are reduced to 19. Only 8 out of 19 polynomials have orders greater than 25 (for $p = 6, M = 25$), and we consider only them. In Table 3, we summarize the obtained results. The generator polynomials of the CRC codes are presented in a hexadecimal notation, i.e. $x^6 + x^5 + x^4 + x^3 + x + 1$ is 73, while their respective reciprocal polynomials are given in the brackets. The weight spectra of the first two codes are not calculated, because they have smaller $S_d$ than the others. We also provided the order $n_c$ of the polynomial, the minimum distance of the CRC codes, and the interval of lengths for which the code has the corresponding minimum distance.

According to our optimization criterion, the CRC code generated by the polynomial 59 is optimal. However, all five codes with a maximum value of $S_d$ have very close
TABLE 3. Results for CRC codes with 6 check symbols.

| g(x)   | n_c | S_d | S_{d_e} | minimum distances |
|--------|-----|-----|---------|------------------|
| 73 (67)| 63  | 24  | -       | -                |
| 6d (5b)| 63  | 24  | -       | -                |
| 61 (43) CRC6 | 63 | 24 | 173 | 3.18...25 |
| 47 (71) | 31  | 32  | 1959    | 4.18...25        |
| 59 (4d) optimal | 31 | 32 | 1956 | 4.18...25 |
| 7b (65) | 31  | 32  | 1966    | 4.18...25        |
| 7d (5f) | 30  | 32  | 1962    | 4.18...25        |
| 4b (96) | 28  | 32  | 1959    | 4.18...25        |

Thus the differences of their \( P_{ue} \) values are negligible and any of them could be a good choice. The code with generator polynomial 47 is also suggested by Philip Koopman in its “Best CRC polynomials” website [13] as the best general-purpose polynomial generating CRC code with 6 check bits (as \( 0 \times 23 \) with HD = 4) and having maximum minimum distance profile. The CRC code with generator polynomial 61 is the proposed in 5G NR CRC6 code and any of the five codes with \( S_d = 32 \) will have much better performance than them. In Figure 2 we present a comparison between the \( P_{ue}' \) of the CRC codes with generator polynomials 61 and 59 for \( \epsilon = 10^{-12} \) and all lengths \( n \) at which the code is supposed to be used. The suggested optimal code is at least 100% better than CRC6 at all lengths. This result is not surprising as CRC6 has a minimum distance 3 for all lengths from 18 to 25 while the optimal CRC code has a minimum distance 4.

FIGURE 2. Comparison of \( P_{ue}' \) of CRC6 and the optimal CRC codes.

B. CRC CODES OF \( P = 11 \)

There are \( 2^{10} = 1024 \) possible choices for generator polynomials of CRC codes of degree 11. After excluding the reciprocal and those of order smaller than 1717, we obtain that the CRC code generated by the polynomial e0f has a maximum \( S_d \). In Table 4 we present the results regarding the CRC codes generated by the e0f polynomial and the proposed in 5G NR one generated by the polynomial e21 (CRC11). The proposed in [2] 93f polynomial is not optimal in this investigation as it has a maximum \( S_d \) only for lengths between 12 and 512. For the rest lengths between 31 and 1717, it is not optimal and thus is not optimal for the whole interval. The CRC code generated by the e0f polynomial possesses significantly better performance than CRC11 in the interval of lengths 55...149 where it has a minimum distance 4 while CRC11 has 3. For example, for \( \epsilon = 10^{-12} \) at length 130 the optimal code is 100% better than CRC11, at length 300 it is 50% better, while above lengths 900 they perform almost equally well, and there are lengths at which CRC11 performs a bit better.

TABLE 4. Results for CRC codes with 11 check symbols.

| g(x)     | n_c | S_d | minimum distances |
|----------|-----|-----|------------------|
| e0f (007)| 1953| 5180| 3.150...1717.431...149 |
| e21 (847) CRC11 | 2047 | 5085 | 3.55...1717.431...53 |

C. CRC CODES OF \( P = 16 \)

For CRC codes with 16 check bits, the maximum \( S_d \) for lengths up to 3840 has the polynomial 1a2eb. In Table 5 the values of \( S_d \) for the polynomials 1a2eb and 11021 (CRC16) are given. The polynomial 1a2eb is suggested by Funk [14] (321353 in his notation) as the polynomial that generates CRC codes with the best undetected error probability performance for the whole range of lengths (17..32767). This polynomial is also suggested by Koopman [13] in his website, as well as in our previous work [2]. Its performance will be much better for lengths up to 109 where it has minimum distance 6 while the proposed polynomial 11021 has minimum distance 4. For example, for \( \epsilon = 10^{-12} \) at lengths 23 and 100 the optimal code is 100% better than CRC16. For lengths between 385 and 875, both polynomials perform almost equally well even though there are lengths at which CRC16 performs a few percent better. After this interval, the suggested polynomial 1a2eb is strictly better than CRC16.

TABLE 5. Results for CRC codes with 16 check symbols.

| g(x)        | n_c | S_d | minimum distances |
|-------------|-----|-----|------------------|
| 1a2eb (1ae8b) | 32767 | 15508 | 4.110...3840.6...109; 8.19...27; 10.17...18 |
| 11021 (10811) CRC16 | 32767 | 15296 | 4.17...3840 |

D. CRC CODES OF \( P = 24 \)

Since all of the three suggested CRC codes with 24 check bits should work for lengths up to 8448 we put \( M = 8448 \).
worse for lengths above 7060 where CRCA has a smaller number of codewords of minimum weight. As suggested by us three codes have better values of $S_d$ they will perform better in general (i.e. when they are used at different lengths) and their slightly worst performance at lengths close to the end of the interval [3848 . . . 8448] will be compensated by their much better performance at shorter lengths.

The suggested in [2] 11175b7 polynomial generates CRC code with the best value of $S_d$ for lengths between 25 and 512. It has order $1195740$ and could be used in the interval [25 . . . 8448] but would have worse performance than the suggested in this work three CRC codes because it has minimum distance 6 only to length 586. This example illustrates once again the necessity to take into consideration the particular conditions (length, bit error rate) at which the code will be used to obtain the best one.

We do not consider the interval [25 . . . 164] since the same code will also be used for the interval [25 . . . 8448]. We only note for completeness that for the interval [25 . . . 164] CRC24C has better performance with $S_d = 924$ than the suggested by us codes where the best $S_d = 802$ is achieved by 118b983. Thus CRC24C is a good choice if it is used only for lengths between 25 and 164.

### VI. CONCLUSION

The error control performance of CRC codes with 6, 11, 16, and 24 check bits for the particular lengths at which the proposed in the technical specification of 5G new radio CRC codes are supposed to be used is investigated. For each number of check bits, optimal codes are suggested and their performance is compared to the performance of the aforementioned proposed codes.

We have found five CRC codes with 6 check bits that have very close error detection performances and any of them could be a good enough, from a practical point of view, choice. Among these five codes is the already suggested by Philip Koopman in its “Best CRC polynomials” website [13] CRC code of 6 check bits. The proposed in our previous work [2] best CRC code of 11 bits has maximum $S_d$ for lengths between 12 and 512, but not for the whole interval of lengths between 31 and 1717 specified in 5G NR. In this work, we found another optimal 11 check bit CRC code. The best CRC code of 16 check bits we have found has already been suggested in three previous works. Funk [14] suggested it as the best shortened linear code with 16 bit redundancy. It also is Koopman’s proposal in the maintained by him website [13] and is obtained by us in [2]. We were not able to find only one CRC code of 24 check bits which is optimal for the whole interval it is proposed to work. It turned out that different codes are optimal at different lengths and we present the three best ones according to our optimization criterion. For example, at length 920 the three optimal codes are between 94% and 100% better than the three suggested in 5G NR CRC24 codes. For the code lengths from the interval [3848 . . . 8448] CRC24A, CRC24B and the suggested by us three codes have minimum distance 4 (see Table 7) and their performance is very close. For example, the optimal codes perform slightly

### TABLE 6. Results for CRC codes with 24 check symbols.

| $g(x)$          | $n_c$   | $S_d$  |
|-----------------|---------|--------|
| 118b933 (1993a31) | 139230  | 35584  |
| 125ae5d (174eb49)  | 6241542 | 35664  |
| 10f66fd (16dde1)   | 294903  | 35548  |
| 1864eb (1be64c3) CRC24A | 8388607 | 34816  |
| 1800063 (18e0003) CRC24B | 8388607 | 33704  |
| 1b2b117 (1d11a9b) CRC24C | 28062  | 31109  |

### TABLE 7. Minimum distances for CRC codes with 24 check symbols.

| $g(x)$          | Minimum distances |
|-----------------|--------------------|
| 118b933 (1993a31) | 4.916...8448; 6.60...915; 8.39...59; 10.29...38; 12.25...28 |
| 125ae5d (174eb49)  | 4.896...8448; 6.64...895; 8.43...63; 10.28...42; 14.25...27 |
| 10f66fd (16dde1)   | 4.881...8448; 6.74...880; 8.42...73; 10.27...41; 14.25...26 |
| 1864eb (1be64c3) CRC24A | 4.542...8448; 6.55...541; 8.34...54; 10.28...33; 12.26...27; 14.25 |
| 1800063 (18e0003) CRC24B | 4.29...8448; 6.25...28 |
| 1b2b117 (1d11a9b) CRC24C | 3.5135...8448; 4.504...5134; 5.182...503; 6.52...181; 7.43...51; 9.32...42; 13.25...31 |
REFERENCES

[1] 5G NR Technical Specification. Accessed: Apr. 11, 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/138200_138299/138212/17.01.00_60/ts_138212v170100p.pdf

[2] T. Baicheva and P. Kazakov, “CRC selection for decoding of CRC-polar concatenated codes,” in Proc. 9th Balkan Conf. Informat., Sep. 2019, pp. 1–5.

[3] W. Peterson and D. Brown, “Cyclic codes for error detection,” Proc. IRE, vol. 49, no. 1, pp. 228–235, Jan. 1961.

[4] W. W. Peterson and E. J. Weldon, Error-Correcting Codes. Cambridge, MA, USA: MIT Press, 1972.

[5] S. B. Wicker, Error Control Systems for Digital Communication and Storage, vol. 1, Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.

[6] A. Vardy, “The intractability of computing the minimum distance of a code,” IEEE Trans. Inf. Theory, vol. 43, no. 6, pp. 1757–1766, Nov. 1997.

[7] T. Fujiwara, T. Kasami, A. Kitai, and S. Lin, “On the undetected error probability for shortened Hamming codes,” IEEE Trans. Commun., vol. COM-33, no. 6, pp. 570–574, Jun. 1985.

[8] J. K. Wolf and R. D. Blakeney, “An exact evaluation of the probability of undetected error for certain shortened binary CRC codes,” in Proc. 21st Century Mil. Commun.-Possible. Conf. Rec. Mil. Commun. Conf., Oct. 1998, pp. 287–292.

[9] G. Castagnoli, S. Brauer, and M. Herrmann, “Optimization of cyclic redundancy-check codes with 24 and 32 parity bits,” IEEE Trans. Commun., vol. 41, no. 6, pp. 883–892, Jun. 1993.

[10] D. Chun and J. K. Wolf, “Special hardware for computing the probability of undetected error for certain binary CRC codes and test results,” IEEE Trans. Commun., vol. 42, no. 10, pp. 2769–2772, Oct. 1994.

[11] P. Kazakov, “Fast calculation of the number of minimum-weight words of CRC codes,” IEEE Trans. Inf. Theory, vol. 47, no. 3, pp. 1190–1195, Mar. 2001.

[12] J. Macwilliams, “A theorem on the distribution of weights in a systematic code,” Bell Syst. Tech. J., vol. 42, no. 1, pp. 79–94, Jan. 1963.

[13] Best CRC Codes. Accessed: Jun. 16, 2020. [Online]. Available: https://users.ece.cmu.edu/~koopman/crc/

[14] G. Funk, “Determination of best shortened linear codes,” IEEE Trans. Commun., vol. 44, no. 1, pp. 1–6, Jan. 1996.

TSONKA BAICHEVA received the M.Sc. degree in computer science from the Technical University of Sofia, Bulgaria, in 1985, and the Ph.D. and Doctor of Science degrees from the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, in 1998 and 2015, respectively.

Currently, she is a Professor with the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences. Her current research interests include coding theory, combinatorics, algorithms, and sequences.

PETER KAZAKOV received the M.Sc. degree in mathematics from Sofia University, Bulgaria, in 1994, and the Ph.D. degree from TU Delft, The Netherlands, in 2000. His Ph.D. thesis was in the area of CRC and spherical codes.

After defended his thesis, he was with SW companies, such as NXP, ST-Ericsson, ASML, and SAP, where he combines programming skills with passion for algorithms. He has also a part time contract with the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, where he continues the research on the topics of the Ph.D. thesis.

MIROSLAV DIMITROV was born in Yambol, Bulgaria, in 1985. He received the B.S. degree in informatics and the M.S. degree in information security from the Faculty of Mathematics and Informatics, Sofia University, and the Ph.D. degree from the Institute of Mathematics and Informatics, Bulgarian Academy of Sciences in 2023. His research interests include cryptology, algorithms, and sequences.

* * *