Some Cryptanalysis of the Block Cipher BCMPQ

V. Dimitrova, M. Kostadinoski, Z. Trajcheska, M. Petkovska and D. Buhov

Faculty of Computer Science and Engineering
"Ss. Cyril and Methodius" University,
Skopje, Macedonia
vesna.dimitrova@finki.ukim.mk, {mile.kostadinoski.mk, zlatka.trajcheska, dbuhov}@gmail.com, marija.petkovska@yahoo.com

Abstract. When designing a new block cipher, the authors must ensure that its design is strong enough for potential attacks or security leaks. That is why at least basic cryptanalysis must be conducted which would point out potential flaws of the crytosystem. In this paper we will discuss some security aspects important for the block cipher BCMPQ - Block Cipher by Matrix Presentation of Quasigroups, for its security and reliability.

Keywords: quasigroup, block cipher, cryptanalysis, avalanche effect

1 Introduction

In [5] the authors introduced a design of a new block cipher, named Block Cipher by Matrix Presentation of Quasigroups (BCMPQ). This cipher is mostly based on quasigroup transformations presented in a matrix form.The basic idea of the design of the block cipher is described in the following.

Quasigroups are algebraic structures whose number is exponentially growing when their order is increasing. Their large number and specific properties make them desirable for cryptographic and coding purposes, and interesting for research in the field of cryptology. Formally, a groupoid $(Q, *)$, where * is binary operation, is called a quasigroup if it satisfies:

$$(\forall\, a, b \in Q)(\exists!\, x, y \in Q)(x * a = b \wedge a * y = b) \tag{1}$$

Now, the quasigroup operation can be presented in matrix form as

$$x * y = m^T + Ax^T + By^T + CAx^T \cdot CBy^T \tag{2}$$

where $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ and $B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$ are nonsingular Boolean matrices and $m = [m_1, m_2]$ is a Boolean vector. There are 4 choices for the matrix C (see [4]), and for this design is chosen the fixed matrix $C = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$. The operation"·"

denotes the dot product, i.e., it is the sum of the products of the corresponding components of the vectors CAx^T and CBy^T.

The encryption/decryption functions of the cipher are built by using e- and d- transformations [2]. Namely, given $a_1a_2 \dots a_n,\ a_i \in Q$, and a fixed element $l \in Q$, called the *leader*, the encryption/decryption functions are defined as:

$e_l(a_1a_2 \dots a_n) = (b_1b_2 \dots b_n) \Leftrightarrow b_1 = l * a_1,\ b_i = b_{i-1} * a_i,\ i \geq 2.$

$d_l(a_1a_2 \dots a_n) = (c_1c_2 \dots c_n) \Leftrightarrow c_1 = l * a_1,\ c_i = a_{i-1} * a_i,\ i \geq 2.$

We note that e_l and d_l are permutations of the set Q^n since the equalities $d_l(e_l(a_1a_2 \dots a_n)) = a_1a_2 \dots a_n = e_l(d_l(a_1a_2 \dots a_n))$ are true for each $a_i \in Q$.

The design of the block cipher is based on three algorithms: round key generation, encryption of single block and its decryption. Also, for the design is used the CBC mode of operation.

There are 144 quasigroups of form (2). Out of them, 128 are chosen and stored in memory as follows:

$$seq\_num \qquad m_1, m_2, a_{11}, a_{12}, a_{21}, a_{22}, b_{11}, b_{12}, b_{21}, b_{22} \tag{3}$$

where *seq_num* is a seven bit number while $m_1, m_2, a_{11}, a_{12}, a_{21}, a_{22},\ b_{11}, b_{12}, b_{21}, b_{22}$ are the bits appearing in the matrix form (2) of the quasigroup operation.

The encryption and decryption algorithms include the use of 16 quasigroups. They are denoted by $Q_1, Q_2, ..., Q_8$ and $T_1, T_2, ..., T_8$ and they are used in different steps. These matrices are determined by using the round key, which is generated out of the secret key and consists of 128 bits.

The key length of 128 bits is distributed in the following way:

- 16 bits for the leaders $l_1, l_2, ..., l_8$ (two bits per each leader)
- 56 bits for the quasigroups $Q_1, Q_2, ..., Q_8$ (7 bits per each quasigroup)
- 56 bits for the quasigroups $T_1, T_2, ..., T_8$ (7 bits per each quasigroup)

The 7 bits designated for each quasigroup are actually the binary representation of the *sequence_number* of the quasigroup (see (3)).

The three algorithms: for round key generation, for encryption of single block and for decryption are presented in [5].

In the next sections we give a partial cryptanalysis of cipher in order to test its security and reliability. Mainly, we will focus on the difference between input and output bits and the strict avalanche criterion (avalanche effect) of the given cipher.

2 Analysis of the input and output bits of the BCMPQ

First we are going to analyze the difference between input and output bits of the BCMPQ, i.e. how the cipher is changing the bits of the input message to obtain the output message. If a cipher changes the input bits with a probability of approximately 50% then the cipher provides good randomization of the bits.

In order to see how the BCMPQ changes the bits of the input messages we made several experiments. In each of the experiments we chose special cases of input messages, encrypt them with the specially chosen secret keys and then compared the encrypted massages with the original messages. Here, we give some results from one of the experiments that were made during this research. In this experiment we chose the input messages to be a periodical strings (for example 111111111111) or inputs that have different first or last bits (for example 011111111 or 1111111100). The secret keys that were used in the experiment are given in Table 1.

Table 1. Secret keys used for encryption of input messages

Key name	Secret key
key 1	101010101010101010101010101010 . . . 1010101010101010101010101010
key 2	000000000000000000000000000000 . . . 0000000000000000000000000001
key 3	111111111111111111111111111111 . . . 1111111111111111111111111111
key 4	010101010101010101010101010101 . . . 0101010101010101010101010101
key 5	110111010100011001111101101010 . . . 0001000111101010101011100001
key 6	001010010001111101000010100101 . . . 1011110101001011011001001000

The results obtained from this experiment are given in Table 2. In each column of the table is presented the change percentage of the bits of the messages, when the appropriate secret key is used.

From the results showed in Table 2 we can see that the change percentage of the bits is mainly around 50%. Therefore, we conclude that using the cipher we obtained good mixing of the bits.

3 Analysis of the strict avalanche criterion of BCMPQ

Next, we give an analysis of the strict avalanche criterion (avalanche effect) of the BCMPQ, i.e. how a change of a single bit in the input message affects the output bits of the encrypted message. For this criterion to be satisfied it requires that whenever a single input bit is changed (from 0 to 1 or from 1 to 0), each of the output bits changes with a probability of approximately 50%. In order to test if the cipher satisfied this criterion, we generated several random secret keys, and using each of these secret keys we encrypted 1000 different randomly generated messages of the same length. Then, we changed a particular bit in the input messages (the bit with the same sequence number in all of the input messages), we encrypted these changed messages with each of the secret keys that we generated and then compared them with the original encrypted messages to see how they differ from each other. We repeated this n times (where n is the length of the input message), so that every single bit in the input messages is changed in the process. The input messages which were used for encryption in the experiments consisted of 64 bits, 128 bits and 192 bits. In Table 3 are shown partial results from one of the experiments, where the length of the input messages was 64 bits.

Table 2. Change percentage of the bits in the messages

Input name	key 1	key 2	key 3	key 4	key 5	key 6
Input 1	50,00%	56,25%	54,69%	48,44%	46,88%	45,31%
Input 2	43,75%	54,69%	50,00%	54,69%	46,88%	43,75%
Input 3	50,00%	46,88%	43,75%	50,00%	53,13%	57,81%
Input 4	54,69%	51,56%	54,69%	42,19%	57,81%	53,13%
Input 5	39,06%	50,00%	48,44%	48,44%	45,31%	54,69%
Input 6	39,06%	40,63%	45,31%	56,25%	51,56%	50,00%
Input 7	50,00%	45,31%	43,75%	56,25%	46,88%	40,63%
Input 8	53,13%	54,69%	45,31%	59,38%	48,44%	46,88%
Input 9	42,19%	43,75%	48,44%	51,56%	54,69%	60,94%
Input 10	67,19%	54,69%	43,75%	46,88%	48,44%	51,56%
Input 11	50,00%	60,94%	50,00%	48,44%	40,63%	46,88%
Input 12	45,31%	54,69%	57,81%	57,81%	34,38%	51,56%
Input 13	53,13%	53,13%	56,25%	54,69%	46,88%	53,13%
Input 14	43,75%	45,31%	45,31%	50,00%	56,25%	57,81%
Input 15	31,25%	60,94%	51,56%	48,44%	46,88%	40,63%
Input 16	42,19%	57,81%	56,25%	59,38%	50,00%	43,75%
Input 17	45,31%	51,56%	56,25%	39,06%	43,75%	56,25%
Input 18	57,81%	48,44%	46,88%	53,13%	53,13%	59,38%
Input 19	56,25%	46,88%	42,19%	34,38%	59,38%	40,63%
Input 20	40,63%	46,88%	45,31%	50,00%	46,88%	42,19%

Table 3. Strict avalanche criterion of the BCMPQ

Input	Output bits									
bits	1	2	3	4	6	7	8	16	32	64
1	51,80%	48,20%	48,10%	47,70%	49,60%	47,40%	52,40%	48,90%	49,30%	52,00%
2	50,00%	52,90%	45,80%	51,20%	48,90%	50,40%	50,90%	52,40%	51,20%	48,70%
3	49,00%	48,90%	50,60%	51,80%	47,80%	49,90%	50,40%	51,70%	49,00%	48,90%
4	50,00%	49,70%	50,30%	49,30%	48,60%	49,30%	50,60%	50,10%	50,40%	49,40%
5	50,30%	50,90%	51,10%	51,30%	49,40%	**50,00%**	48,40%	48,50%	51,90%	47,90%
6	51,80%	50,50%	50,50%	50,50%	50,00%	50,50%	51,20%	50,90%	49,40%	50,20%
7	50,40%	52,50%	51,80%	50,20%	48,50%	51,10%	50,20%	49,40%	50,30%	49,70%
8	50,80%	49,00%	50,50%	49,10%	50,00%	49,20%	50,30%	51,80%	47,00%	50,50%
15	53,60%	50,70%	49,40%	51,00%	49,60%	51,50%	50,60%	49,40%	50,20%	51,80%
16	50,60%	50,50%	51,80%	49,10%	49,80%	49,00%	48,80%	49,90%	49,40%	48,50%
17	47,40%	49,80%	49,20%	49,80%	47,00%	48,00%	50,90%	47,80%	49,70%	47,90%
31	50,50%	51,00%	52,70%	49,70%	51,70%	49,80%	49,80%	50,90%	50,60%	51,30%
32	48,80%	49,30%	45,80%	51,00%	49,40%	52,10%	52,00%	49,10%	50,80%	49,00%
33	51,60%	50,30%	49,80%	47,20%	50,20%	51,60%	52,10%	50,00%	48,90%	45,70%
62	50,00%	50,20%	47,10%	49,30%	49,60%	50,80%	47,30%	52,30%	51,30%	49,50%
63	47,90%	51,70%	52,90%	48,50%	49,50%	48,60%	49,80%	50,90%	54,50%	51,00%
64	48,40%	52,80%	50,60%	52,60%	49,20%	51,20%	50,40%	48,80%	49,90%	48,80%

In Table 3 are shown the change percentages of several bits of the encrypted messages, when a particular bit of the input messages is changed. Each cell of the table represents the change percentage of the j^{th} output bit, when the i^{th} bit of the input messages is changed, where i is the row number and j is the column number of the table. For example, we can see from the table that when the 5th bit of the input messages is changed, then the 7th output bit changed in 50% or half of the encrypted messages (bold value in the table).

For better presentation, we give a graphical presentation. In Fig. 1 is shown the change percentage of all output bits, from the previous experiment, when the 16th input bit is changed. Similar graphical results were obtained when other than the 16th bit of the input messages was changed.

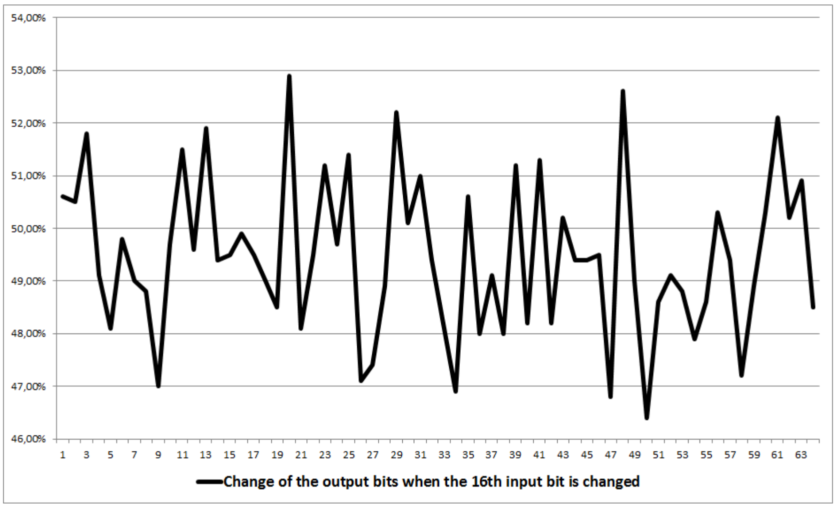

Fig. 1. Graphical presentation of the change percentage of the output bits when the 16th bit is changed

We are also providing a graphical presentation of the change percentages of a single output bit, when all of the input bits are changed, which is given in Fig. 2. The results shown in Fig. 2 are also derived from the previously discussed experiment and refer to the change percentage of the 16thoutput bit. As expected, similar graphical results were obtained for the other output bits.

From Table 3, Fig. 1 and Fig. 2 we conclude that, when an arbitrary bit of the input messages is changed, then every bit of the output is changed with the probability between 44% and 56%. This means that if a single bit is changed in all of the 1000 input messages, then each of the output bits will change in approximately half of the encrypted messages. This result implicates that the cipher satisfies the strict avalanche criterion.

Similar results were obtained from the other experiments, where different secret keys were used for encryption and different messages were encrypted. This can be seen in Fig. 3, where are shown the results from one of the other experiments. In this experiment, one input message was encrypted with random secret key, then each of the input bits was changed, one at a time, and the changed messages were encrypted and compared to the original encrypted message. The percentages of changed output bits, when each input bit was changed, are shown in Fig. 3. Note that every secret key and message, that were used in all of the experiments were different from each other and also randomly generated.

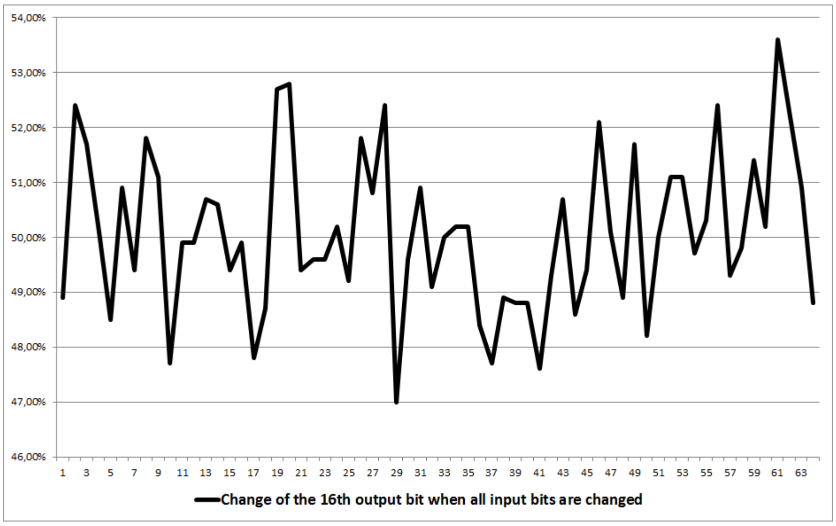

Fig. 2. Graphical presentation of the change percentage of the 16^{th} bit when all inputs bits are changed

The results from the experiments where 128 bit and 192 bit input messages were used, slightly differed from the previous. In fact, in these experiments, if the change was made in a bit in the second or third block of the input messages, i.e. if the change was made after the 63^{rd} bit of the input messages, than the change only affected bits of the same block and all of the consecutive blocks of the encrypted messages. The bits of the previous blocks were not affected at all, and remain the same as before the change in the input message was made. This was actually expected and it is a result of the structure of the cipher, which uses Cipher-block chaining (CBC) mode of operation. In CBC mode of operation, the bits of each block of the input message are XOR-ed with the bits of previous output block (with exception of the first block which is XOR-ed with a random initialization vector IV), before being encrypted. This means that if we make a change in one bit of the input message, then we can expect a change in the bits of the same or the consecutive blocks of the encrypted message, because a change in the bits of the previous blocks of the encrypted message will occur with 0 probability. If we ignore the change percentage of the bits of previous blocks of the encrypted messages, then we can say that the change of each bit in the input message results with a change in the bits of the encrypted messages with probability of approximately 50%. This is same as in the experiments with 64 bit input messages.

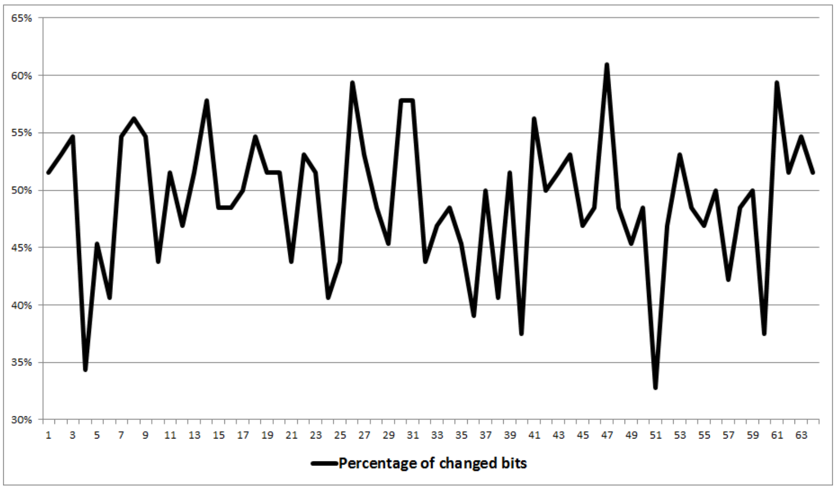

Fig. 3. Percentages of changed bits in the output message when each bit in the input message was changed

In general, we conclude that a small change in the input message results with a significant change in the encrypted message with the cipher, which is a desired feature of a high-quality block cipher. All of the experiments that were made showed that a change of each bit of the input message affects the output bits with approximately 50% probability, which shows that the cipher has good randomization. A good randomization is necessary, because it prevents predictions about the input message, when only the encrypted message is known.

4 Avalanche effect with a change of two input bits

In this section we are examining the change percentage of each of the output bits when two bits of the input message are changed. In order to do this, we made encryptions of 64 input messages, then in each of the these messages we changed two bits on different positions and encrypted them with the appropriate keys. After that we compared the originally encrypted messages with the changed encrypted messages to see how they differ from each other. The obtained results are shown on Fig. 4.

As we can see in Fig. 4 the percentages vary, but they are mostly between 45% and 55%, which is good, because in the ideal case they would be 50%.

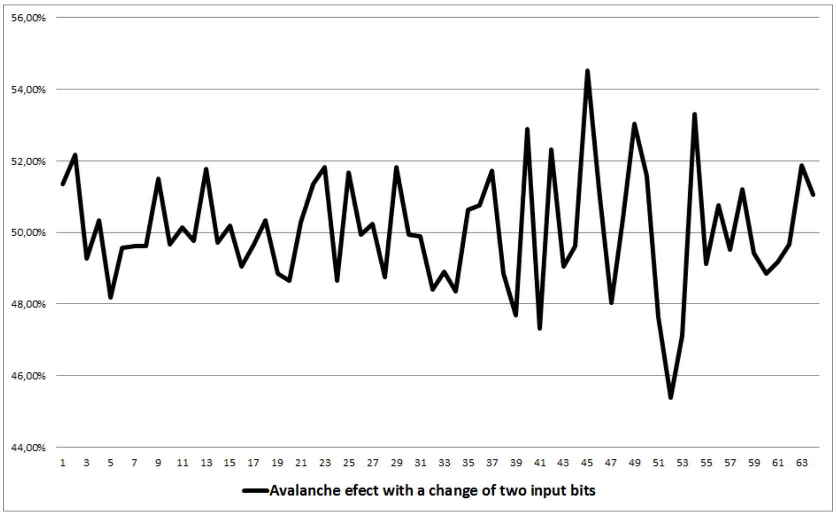

Fig. 4. Statistics of the bit changes in the encrypted message when 2 bits are changes

5 Conclusion

In this paper we made partial cryptanalysis of the block cipher BCMPQ. We analyzed the change percentage of the bits in the input messages and concluded that it is approximately 50%. We also analyzed the Strict avalanche criterion of the BCMPQ and the results showed that when random input bit is changed, then the cipher changes each output bit with a probability of approximately 50%. Therefore, we concluded that the cipher satisfies this criterion. We also extended the analysis of the avalanche effect, with the change of two input bits and the results showed that even when two random input bits are changed, the change percentage of the output bits again is approximately 50%, which was a desired result for the cipher.

Although the experimental results that were made showed that the BCMPQ has good randomization, a mathematical proof of the same is still required in order to be absolutely sure that the cipher satisfies the randomness, which is a future work for the authors of BCMPQ.

The obtained results are only a part of the many other attacks that exist, like differential and linear cryptanalysis. They are still open problems for further investigation.

References

1. Dimitrova, V: Quasigroup Processed Strings, their Boolean Representations and Application in Cryptography and Coding Theory. PhD Thesis (2010) Ss. Cyril and Methodius University, Skopje, Macedonia

2. Markovski S., Gligoroski D., Bakeva V., Quasigroup String Processing: Part 1, Contributions, Section of Mathematical and Technical Sciences,, Macedonian Academy of Sciences and Arts, XX 1-2, 1999, pp. 13-28
3. Mileva, A. (2010). Cryptographic Primitives with Quasigroup Transformations.
4. Siljanoska, M., Mihova, M.,Markovski, S.: Matrix Presentation of Quasigroups of order 4, The 10th Conference for Informatics and Information Technology (CIIT 2013), Bitola, 2013
5. Markovski, S., Dimitrova, V., Trajcheska, Z., Petkovska, M., Kostadinoski, M., Buhov, D.: Block cipher defined by matrix presentation of quasigroups, The 11th Conference for Informatics and Information Technology (CIIT 2014), Bitola, 2014 (in print)