JJEE
Volume 11 | Number 1 | March 2025 | Pages 168-179

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

Data Shuffling and Data Blocks Shuffling Method for Digital Data Signals Protection

Hisham O. Alrawashdeh[1*]

[1]Electrical Power and Mechatronics Engineering Department, College of Engineering, Tafila Technical University, Tafila, Jordan
E-mail: hisham@ttu.edu.jo

Received: Aug 06, 2024 *Revised: Oct 18, 2024* *Accepted: Oct 27, 2024* *Available online: Nov 28, 2025*

***Abstract*—** In this paper, a new method for digital data cryptography is proposed to increase the speed of data cryptography, simplify the processes of data encryption - decryption, and to strengthen the degree of data protection. The proposed data shuffling and data blocks shuffling (DS_DBS) method is able to process messages, gray images, color images and digital speech file using the same encryption and decryption functions; changing the digital data type will not require any changes in the aforesaid functions. The proposed DS_DBS method allows data blocking (block size will be variable), and it is determined by the private key. Data encryption is applied by simple data shuffling and data blocks shuffling, while data decryption is applied by shuffling back the data blocks and shuffling back the data. These shuffling operations replace the complex logical and arithmetic operations used in other existing methods of data cryptography. The shuffling and shuffling back operations are implemented based on two secret indices keys, obtained by running two chaotic logistic map models. The proposed method utilizes a long private key with a variable length that depends on the selected number of crypto phases. It can be implemented in one or more phases; each phase is a function call to execute the encryption-decryption functions with the associated inputs. Using more than a phase, increases the security level by using longer private key, and each phase will be independent. The encrypted-decrypted result of each phase can be taken as a final result. Implementing the proposed DS_DBS method - using various data types – and examining its speed reveal not only the enhanced speed of data cryptography, but also the better quality and sensitivity of the proposed method compared to those of existing state of the art cryptography methods.

***Keywords*—** Cryptography; Data shuffling; Data blocks shuffling; Data encryption – decryption.

1. INTRODUCTION

Regardless of its type, digital data may be secret, private, or may be used in a computer application, which requires a high level of security; so protecting digital data from being hacked is a vital issue [1-10]. One on the most common and widely used technique to protect digital data is data cryptography [11-15].

Data cryptography can be implemented by means of encryption and decryption functions as shown in Fig. 1. Encryption function is usually executed by the data sender and it treats the input data with the private key (PK) to produce a cipher (encrypted) data, while the decryption function usually executed by the data receiver, and it treats the cipher data and the PK to produce a decrypted data [16-20].

A crypto method will be considered as a good one if it satisfies the following requirements [21-25]:

- Speed: the method must provide a high speed of data encryption and data decryption, and the throughput of data encryption (K bytes encrypted per second) must be high. Also,

* Corresponding author
DOI: https://doi.org/10.5455/jjee.204-1722936064

the decryption throughput must be high, this can be achieved by minimizing both of the encryption and decryption times [26-28].

- Security: the method must provide a high level of security, the PK must be as long as possible; the longer PK will be capable of providing a huge key space able to resist hacking attacks. The produced decrypted data must be sensitive to any changes in the PK as well.
- Quality: the encrypted data must be totally destroyed, i.e., the encryption function must produce a damaged encrypted data; the mean square error (MSE) measured between the source data and the encrypted data must have a high value, while the value of peak signal to noise ratio (PSNR) measured between them must be low. The produced decrypted data must be the same as the source data, the value of MSE measured between them must equal zero, while the PSNR value must be infinite.
- Flexibility: the method must be capable of handling any digital data type (message, gray image, color image, digital speech file (DSF)); changing the data type must not require any changes in the encryption and decryption functions.
- Simplicity: the method must be simple and easy to program. This can be achieved by simplifying the key generation function, the encryption function and the decryption function. The number of used rounds must be reduced and the sequence of required logical and arithmetic operations must be reduced or eliminated [29-32].

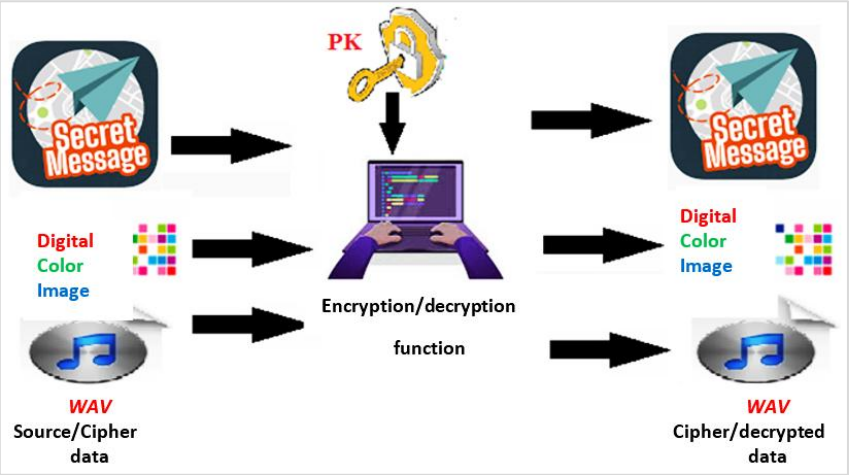

Fig. 1. Crypto method diagram.

Digital data types share an important feature; each digital data can be represented by one row matrix. Message (short or long) is a set of characters organized in one row matrix. DSF is a set of samples organized in one column matrix (mono speech) or organized in two column matrixes (stereo speech); the speech matrix can be easily reshaped to one row matrix.

Gray image is a set of gray colors organized in 2D matrix, while color image is a set colors values organized in 3D matrix, and it is also easy to reshape the 2D and 3D matrices into one row matrix. Doing this the encryption and decryption functions will deal with one row matrix, and it will be easy to handle any data type.

2. RELATED WORKS

A lot of methods were introduced for data cryptography, in this paper we will divide these methods into two groups as follows:

- Slow methods: these methods provide a speed up to 180 kB such as DES, AES, BF, DNA and BDNA.
- Good moderate speed methods: these methods provide up to 900 kB/s such as chaotic and hybrid methods proposed in.

The speed of the proposed data shuffling and data blocks shuffling (DS_DBS) method speed will be compared with the speeds of these methods to show how the proposed method will enhance the speed of data cryptography by providing a good speed up compared to other existing method (speed up equals encryption time of other method divided by the encryption time of the proposed method).

The existing methods of data cryptography share some features, some of them may be considered as a weal points, which require enhancement, following are the main features of these method:

- Data blocking: the data to be encrypted-decrypted must be divided into blocks. The block size is small and fixed. For large data, the number of blocks will be a great one, thus extra time will be required to encrypt-decrypt the data. The proposed DS_DBS method will allow data blocking and the block size will be variable; it will be determined by the private key.
- Number of rounds: each method requires several rounds, each round treats the same data block, and the result must be taken from the last round, each round require a secret key. The proposed DS_DBS method will not require any rounds, it will use one or more phases. Each phase will be independent, and using more than one phase will raise the security level and making the PK longer.
- PK length: the existing methods use PK with fixed length falling between 56 bits in DES method and 448 bits for BF method). The proposed method will use variable length private key, and the length will depend on the number of selected phases (320 bits for each phase).
- Simplicity: Most of the existing methods of data cryptography required a complex of logical and arithmetic operations to implement each round and to generate the required secret keys, this will make programming these methods difficult, the proposed DS_DBS method will eliminate this sequence by replacing it with simple shuffling operations.

3. THE PROPOSED DS_DBS METHOD

The proposed DS_BDS method have the following features:

- The encryption and decryption functions are generalized; they can be used to process any type of data, the data to be passed to each function is a one row matrix data (see Figs. 2 and 3).
- The encryption-decryption processes can be implemented using one or more phases, each function call (calling the encryption or decryption function will be used as a phase), the number of phases in the decryption process must equal the number of phases in the encryption process, but in inverse way (see Fig. 3).
- Each function uses it is own PK, which will contain the number of blocks and two sets of chaotic logistic map parameters (r and x); these two pairs will be used to run two chaotic logistic map model to generate two indices keys (the PK length will be variable and the number of blocks and the block size will be also variable).

- The encryption function will encrypt the data by shuffling the data blocks first, then shuffling the contents of each data block, while the decryption function will shuffle back the contents of each block, then shuffle back the data blocks.

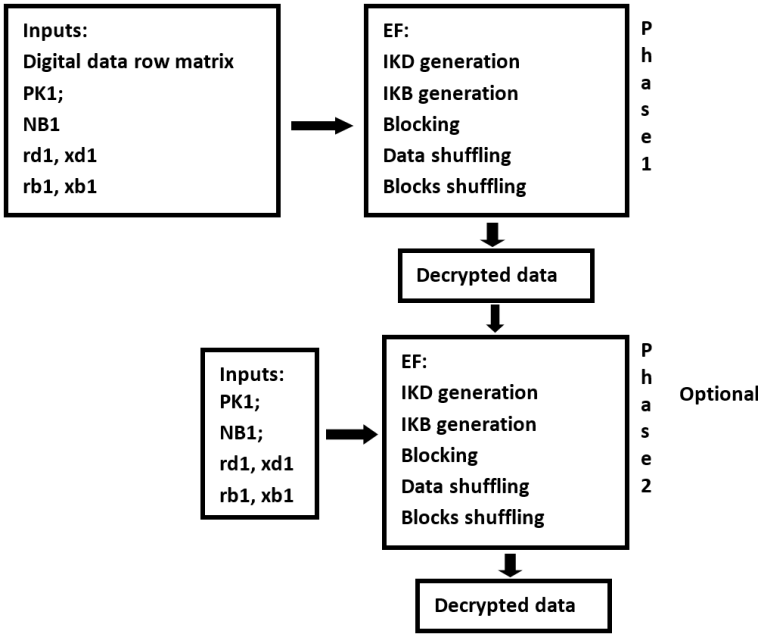

Fig. 2. Encryption function of the generalized DS_BDS method.

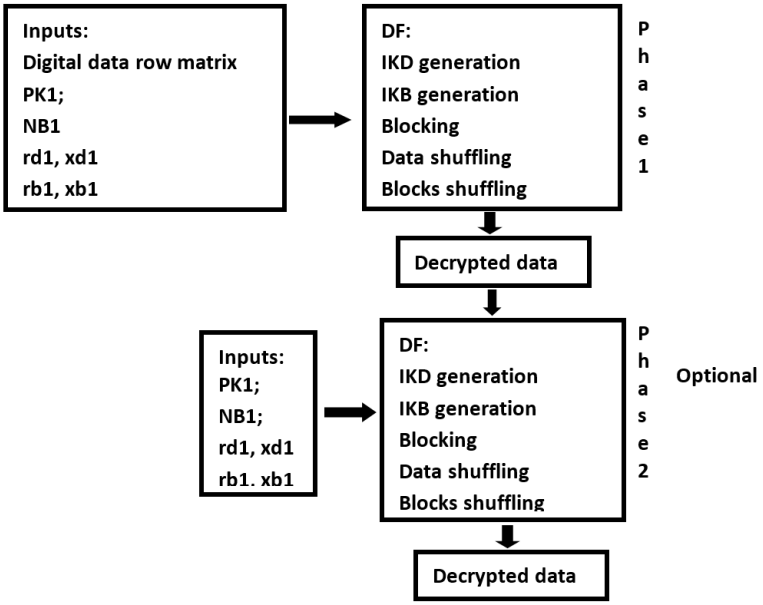

Fig. 3. Decryption function of the generalized DS_BDS method.

3.1. Secret Indices Key Generation

This task will be implemented by both the encryption and decryption functions. Two pairs of the chaotic parameters will be used. The first pair with number of blocks will be used to run the first chaotic logistic map model (CLMM) to get a chaotic key (CK). This CK will be sorted to form the indices key (IK) to be used for data blocks shuffling (IKD), the second pair of chaotic parameters with the block size (BS) will be used to run the second CLMM to generate another CK, this key will be sorted to form the second indices key (IKB) to be used to shuffle the contents of each data block.

Generating an IK is a simple procedure; each CLMM will use the chaotic logistic equation [3-8] to get the CK, Fig. 4 shows an example of how this process can be easily implemented:

```
r= 3.77; x=0.12;          % chaotic pair
 for i=1:10               % 10 is the key length
x=x*r*(1-x);              % chaotic logistic equation
CK(i)=x;
End
[qq IK]=sort(CK);         % IK is the indices key
CK=
0.3981  0.9034  0.3291  0.8324  0.5259  0.9400  0.2127  0.6314  0.8774  0.4056
IK=
                          ↓ Sorting
   7    3    1   10    5    8    4    9    2    6
```

Fig. 4. Example of IK generation

The generated IK is very sensitive to selected values of the PK components, any minor change in one or more values will lead to change in the IK, and thus will change the results of encryption-decryption. As shown in Figs. 3 and 4 the PK of the proposed method using two phases will contain 10 parameters (5 for each phase); thus the PK length will equal 640 bits (10x64). This key length will provide a huge key space - as seen from Eq. (1) - capable to resist hacking attacks.

$$keyspace = 2^{640}$$
$$= 4.562440617622195218641171605 73 \times 10^{192} \ combinations \quad (1)$$

3.2. Data Shuffling

The data shuffling operation is a simple operation and it will be used to get data using the contents of IK as an index, while the shuffling back operation will use the IK to get the data according to the data with index pointing to the smallest index (get smallest first), Fig. 5 illustrates an example of implementing these operations.

```
IK=
    7    3    1    10    5    8    4    9    2    6
Data =
   73  113   4    14  236   83   50  157   39   33
a1 = data;
for i= 1: 10                          Shuffling operation:
c= IK(i);                             Get data according to IK contents
a1(i)=data(c);                        ↓
end
a1 =
   50    4   73   33   236   157   14   39   113   83
a2 = a1;
for i= 1: 10                          Shuffling back operation
c= IK(i);                             Get the smallest (in IK) first
a2(c)=a1(i);                          ↓
end
a2 =
  73   113   4    14   236   83    50   157   39   33
```

Fig. 5. Example of shuffling and shuffling back implementation.

The proposed DS_DBS method will use two simple tasks to implement digital data encryption-decryption:

- Digital data shuffling: this task will be used to divide the data into equal blocks; then these blocks will be shuffled in the encryption function based on the contents of generated indices key (IKM). In the decryption functions the blocks will be shuffled back.
- Data blocks shuffling: this task - in the encryption function - will shuffle the data items within each block based on the contents of generated second indices key (IKB), while in the decryption function the data items of each data block will be shuffled back based on the contents of IKB.

Shuffling operations can be easily implemented. Fig. 6 illustrates an example of message shuffling, while Fig. 7 illustrate an example of data shuffling back in the decryption phase.

3.3. Encryption/ Decryption Functions

The encryption function of the proposed method as shown in Fig. 2 performs three tasks: key generation, data blocks shuffling and shuffling the contents of each data block. Fig. 8 shows the description of these tasks; it illustrates the inputs, generated output and the sequence of operations required to apply data encryption.

The decryption function of the proposed method as shown in Fig. 3 performs three tasks: key generation, shuffling back the contents of each data block and shuffling back the data blocks. Fig. 9 shows the description of these tasks; it illustrates the inputs, generated output and the sequence of operations required to apply data decryption).

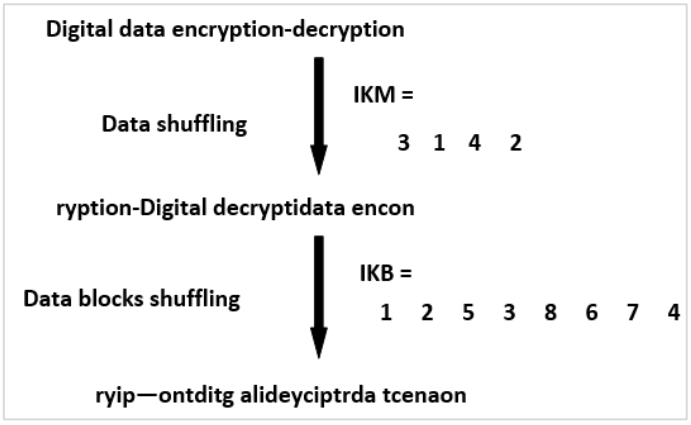

Fig. 6. Digital data encryption (one phase)

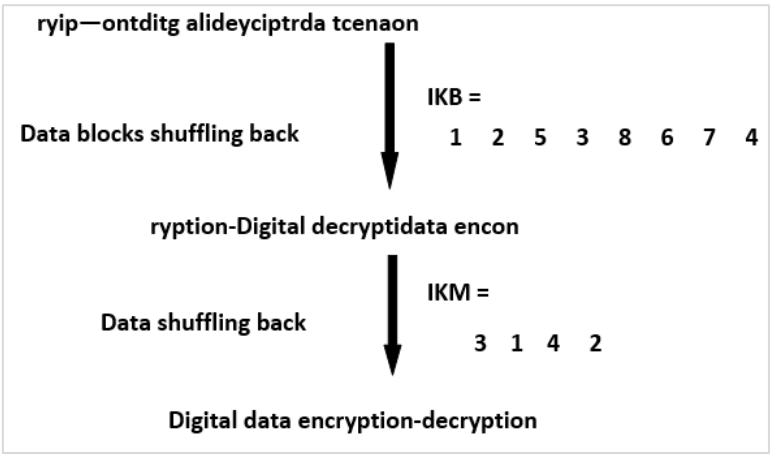

Fig. 7. Digital data decryption (one phase).

```
function [ED]=MS_BS(m,NB,rm,xm,rb,xb)
L=length(m);BS=fix(L/NB);
for i=1:NB
   xm=xm*rm*(1-xm);   CKM(i)=xm;
end
[qq IKM]=sort(CKM);
for i=1:BS
   xb=xb*rb*(1-xb);   CKB(i)=xb;
end
[qq IKB]=sort(CKB);
a1=m;
for i=1:NB
   c=IKM(i);
   a1((i-1)*BS+1:i*BS)=m((c-1)*BS+1:c*BS);
end
a2=a1;
for i=1:NB
   b=a1((i-1)*BS+1:i*BS);   for j=1:BS
      c=IKB(j);
      b1(1,j)=b(1,c);
   end
  a2((i-1)*BS+1:i*BS)=b1;
end
ED=a2;
end
```

Fig. 8. The encryption function.

```
function [DD]=MS_MB_R(m, NB, rm,xm,rb,xb)
L=length(m);BS=fix(L/NB);
for i=1:NB
   xm = xm*rm*(1-xm);   CKM(i)=xm;
end
[qq IKM]=sort(CKM);
for i=1:BS
   xb=xb*rb*(1-xb);   CKB(i)=xb;
end
[qq IKB]=sort(CKB);a1=m;
for i=1:NB
   b=m((i-1)*BS+1:i*BS);
   for j=1:BS
      c=IKB(j);      b1(1,c)=b(1,j);
   end
  a1(1,(i-1)*BS+1:i*BS)=b1;
end
a2=a1;
for i=1:NB
   c=IKM(i);
   a2((c-1)*BS+1:c*BS)=a1((i-1)*BS+1:i*BS);
end
DD=a2;
end
```

Fig. 9. The decryption function.

4. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed DS_BDS method was implemented using a PC having an Intel Core i5 CPU@2.5 GHz and 4 GB RAM. The speed of the proposed method was tested; a set of color images were selected, the following private key was used:

PK:

r=[3.6785 3.8993 3.5830 3.9852];
x=[0.7953 0.0081 0.4110 0.6884];
NB1=150;NB2=200;

Two phases of cryptography were used, the encryption time was measured and the throughput was calculated. Table 1 shows the obtained results.

Table 1. Speed parameters of the proposed method using color images with two phases of cryptography.

Color image dimension	Color image size [byte]	Encryption time [s]	Encryption throughput [kB/s]
151x333x 3	150849	0.0570	2584.4
240x240x3	172800	0.0670	2518.7
360x480x3	518400	0.2320	2182.1
600x1050 x 3	1890000	0.6850	2694.5
1144x 1783x 3	6119256	3.1950	1870.4
Average		0.8472	2370.0

As show in Table 1 the proposed method provides a good speed, the average throughput was equal 2370 K bytes. Compared to the fast group of existing methods speeds [3-8], the proposed method speed up the process of data cryptography by increasing the throughput of data cryptography (see Table 2).

Table 2. Comparison between the proposed and reported – in literature - fast existing methods.

Ref.	Method	Throughput [kB/s]	Speed up of PCS method
[3]	Hybrid	888.8867	2.6663
[4]	Chaotic	638.4082	3.7124
[5]	Chaotic	911.0352	2.6014
[6]	Mixed DNA and chaotic	360.4102	6.5758
[7]	Hybrid	384.9609	6.1565
Proposed	DS_DBS (chaotic)	2370.0	1.0000

Using one phase in the proposed method will enhance the speed. The previous images were implemented and Table 3 shows the obtained speed results. As depicted in Table 3, using one phase of cryptography increased the throughput from 2370.0 kB/s using two phases of cryptography to 4510.5 kB/s using one phase of cryptography.

Using more than one phase will not affect the quality of the method. Iit negatively affects the speed, but it is required if there is a need to increase the security level.

The quality of the proposed method was tested and the obtained results showed that this method satisfied the quality requirements suing any type of digital data. Figs. 10 - 13 are example outputs that prove the quality of the proposed DS_DBS method of digital data cryptography.

The obtained values of MSE and PSNR were always acceptable, and the values of MSE - calculated between the source and the encrypted data - were always high, while the values of

PSNR were always low. The values of MSE calculated between the source and the decrypted data were always zero, while the values of PSNR were always infinite.

Table 3. Speed parameters of the proposed method using color image with one phase of cryptography.

Color image dimension	Color image size [byte]	Encryption time [s]	Encryption throughput [kB/s]
151x333x 3	150849	0.0320	4603.5
240x240x3	172800	0.0380	4440.8
360x480x3	518400	0.1200	4218.7
600x1050 x 3	1890000	0.3470	5319.0
1144x 1783x 3	6119256	1.5050	3970.7
Average		0.4084	4510.5

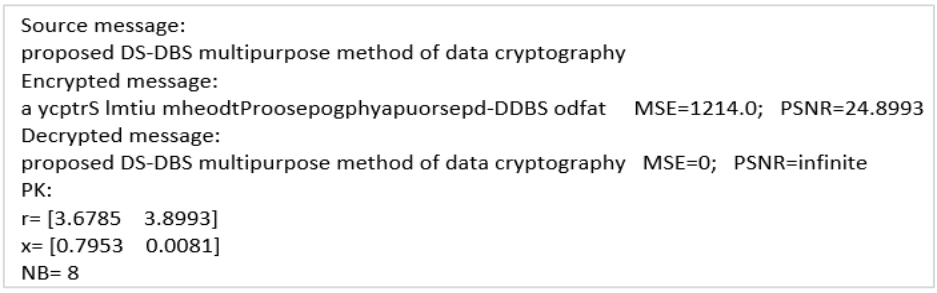

Fig. 10. Quality of messages.

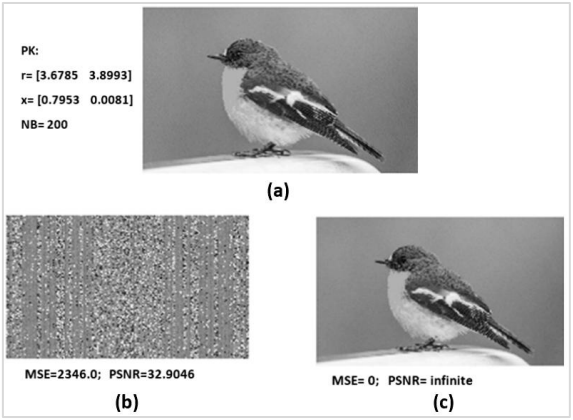

Fig. 11. Gray image quality: a) source image; b) encrypted image; c) decrypted image.

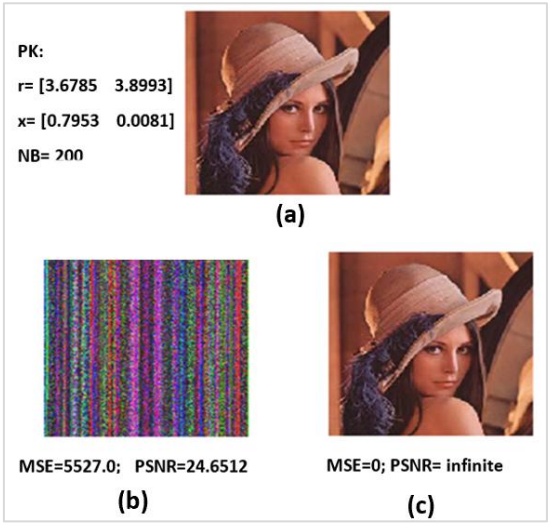

Fig. 12. Color image quality: a) source image; b) encrypted image; c) decrypted image.

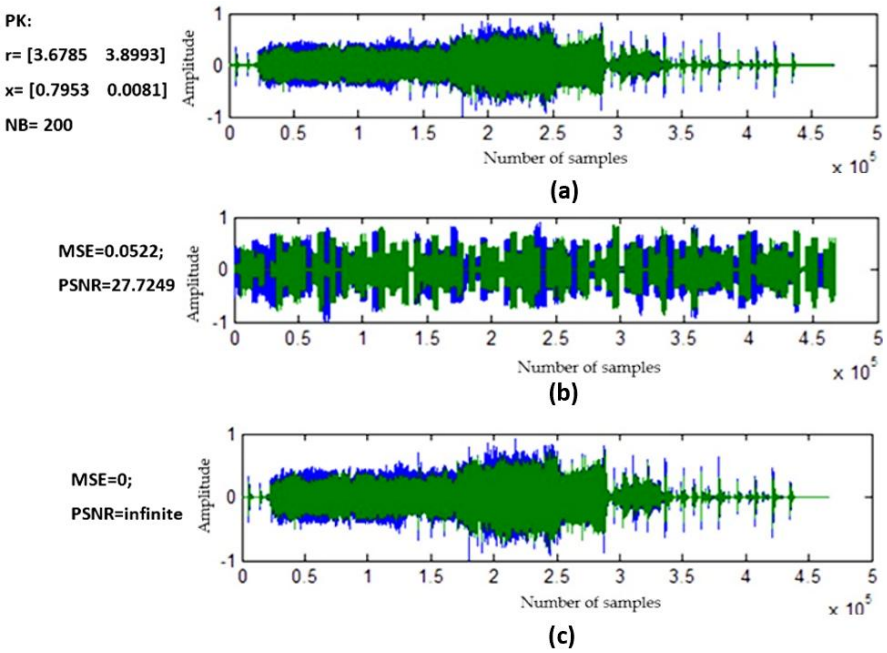

Fig. 13. DSF quality: a) source DSF; b) encrypted DSF; c) decrypted DSF.

5. CONCLUSIONS AND FUTURE WORK

A multipurpose, multiple phases, highly secure and fast method of data cryptography was introduced. The method was efficiently used to encrypt-decrypt messages, images and digital speech file. It uses independent phases to apply data encryption-decryption; the increased number of phases was used to raise the security level of the data, keeping the method fast. The proposed method used a PK with a length of 320 bits, that was increased to 640 bits when using two phases. The private key provided a huge key space capable of resisting hacking attacks, and the produced outputs were sensitive to the selected values of the PK. The proposed method used data blocking with variable unlimited block sizes, thus eliminating the rounds and the sequence of logical operations used in other methods of data cryptography. The proposed method used a simple task to generate the required two indices keys for each phase; these keys were generated by running two simple chaotic logistic map models. The data encryption-decryption was applied using simple data blocks shuffling and simple blocks contents shuffling based on the contents of the generated indices keys.

The proposed method was tested and implemented using various messages, various images and various DSFs. The obtained results proved the quality of the proposed method. The speed results of the proposed method were compared with other existing fast method speeds and the results showed that the proposed method provides a good speed up.

As a future work, this method will be deployed in social media tools such as mobile, whatsapp, and email to protect the sent data, and its performance will be tested and evaluated.

REFERENCES

[1] A. Nadeem, M. Javed, "A performance comparison of data encryption algorithms," International Conference on Information and Communication Technologies, 2005, doi: 10.1109/ICICT.2005.1598556.

[2] W. Stallings, *Cryptography and Network Security Principles and Practices*, Prentice Hall, 2005.

[3] K. Muralidhar, R. Sarathy, "Data shuffling-a new masking approach for numerical data," *Management Science*, vol. 52, no. 5, pp. 658-670, 2006, doi: 10.1287/mnsc.1050.0503.

[4] M. Aqel, Z. Alqadi, I. El Emary, "Analysis of stream cipher security algorithm," *Journal of Information and Computing Science*, vol. 2, no. 4, pp. 288-298, 2007.

[5] S. Jamel, M. Deris, "Diffusive primitives in the design of modern cryptographic algorithms," International Conference on Computer and Communication Engineering, 2008, doi: 10.1109/ICCCE.2008.4580696.

[6] A. Mousta, Z. Alqadi, "A practical approach of selecting the edge detector parameters to achieve a good edge map of the gray image," *Journal of Computer Science*, vol. 5, no. 5, pp. 355-362, 2009.

[7] A. Zaidan, A. Majeed, " High securing cover-file of hidden data using statistical technique and AES encryption algorithm," *World Academy of Science Engineering and Technology*, vol. 54, pp. 468-479, 2009, doi: 10.5281/zenodo.1055032.

[8] D. Salama, A. Minaam, H. Abdual-kader, M. Hadhoud, "Evaluating the effects of symmetric cryptography algorithms on power consumption for different data types," *International Journal of Network Security*, vol. 11, no. 2, pp. 78-87, 2010.

[9] E. William, C. Barker, ''Recommendation for the triple data encryption algorithm (TDEA) block cipher,'' *NIST Special Publication*, vol. 2004, p. 800-67, 2004, doi: 10.6028/NIST.SP.800-67r2.

[10] A. Kaushik, M. Barnela, A. Kumar, "Keyless user defined optimal security encryption,'' *International Journal of Computer and Electrical Engineering*, vol. 4, no. 2, pp. 2-6, 2012, doi: 10.7763/IJCEE.2012.V4.458.

[11] K. Mandal, C. Parakash, A. Tiwari, ''Performance evaluation of cryptographic algorithms: DES and AES,'' IEEE Students' Conference on Electrical, Electronics and Computer Science: Innovation for Humanity, 2012, doi: 10.1109/SCEECS.2012.6184991.

[12] M. Ebrahim, S. Khan, U. bin Khalid, "Symmetric algorithm survey: a comparative analysis," *International Journal of Computer Applications*, vol. 61, no. 20, pp. 12-19, 2013, doi: 10.48550/arXiv.1405.0398.

[13] A. Alshahrani, S. Walker, "Implement a novel symmetric block cipher algorithm," *International Journal on Cryptography and Information Security*, vol. 4, no. 4, pp. 1-11, 2014, doi: 10.5121/ijcis.2014.4401.

[14] S. Manku, K. Vasanth, ''Blowfish encryption algorithm for information security," *ARPN Journal of Engineering and Applied Sciences*, vol. 10, no. 10, pp. 4717-4719, 2015.

[15] J. Zhenjun, R. Sun, "Multiple-image encryption with bit-plane decomposition and chaotic maps," *Optics and Lasers in Engineering*, vol. 80, pp. 1–11, 2016, doi: 10.1016/j.optlaseng.2015.12.004.

[16] N. Kumar, P. Chaudhary, ''Performance evaluation of encryption/decryption mechanisms to enhance data security,'' *Indian Journal of Science and Technology*, vol. 9, no. 20, pp.1-10, 2016, doi: 10.17485/ijst/2016/v9i20/70417.

[17] Z. Hercigonja, D. Gimnazija, C. Varazdin, ''Comparative analysis of cryptographic algorithms and advanced cryptographic algorithms,'' *International Journal of Digital Technology & Economy*, vol. 1, no. 2, pp. 1-8, 2016, doi: 10.4236/jis.2020.113009.

[18] V. Palagushin, A. Khomonenko, ''Evaluation of cryptographic primitives security based on proximity to the Latin square,'' Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology, 2016, doi: 10.1109/FRUCT-ISPIT.2016.7561537..

[19] P. Patil, P. Narayankar, D. Narayan, S. Meena, "A comprehensive evaluation of cryptographic algorithms: DES, 3DES, AES, RSA and Blowfish," *InProcedia Computer Science*, vol. 78, pp.617-624, 2016, doi: 10.1016/j.procs.2016.02.108.

[20] J. Nadir, A. Abu Ein, Z. Alqadi, "A technique to encrypt-decrypt stereo wave file," *International Journal of Computer and Information Technology*, vol. 5, no. 5, pp. 465-470, 2016.

[21] F. Maqsood, M. Ahmed, M. Ali, M. Shah, "Cryptography: comparative analysis for modern techniques," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, 2017, doi: 10.14569/IJACSA.2017.080659.

[22] M. Mushtaq, U. Akram, I. Khan, S. Khan, A. Shahzad, A. Ullah, "Cloud computing environment and security challenges: a review," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, pp. 183-195, 2017, doi: 10.14569/IJACSA.2017.081025.

[23] F. Maqsood, M. Ali, M. Ahmed, M. Shah, ''Cryptography: a comparative analysis for modern techniques,'' *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, pp. 442-448, 2017, doi: 10.14569/IJACSA.2017.080659.

[24] A. Al-Qaisi, S. Khawatreh, A. Sharadqah, Z. Alqadi, ''Wave file features extraction using reduced LBP,'' *International Journal of Electrical and Computer Engineering*, vol. 8, no. 5, pp. 2780-2787, 2018, doi: http://doi.org/10.11591/ijece.v8i5.pp2780-2787.

[25] M. Sohal, S. Sharma, "BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing," *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 1, 2018, https://doi.org/10.1016/j.jksuci.2018.09.024.

[26] Z. Hua, Y. Zhou, H. Huang, "Cosine-transform-based chaotic system for image encryption," *Information Sciences*, vol. 480, pp. 403–419, 2019, doi: 10.1016/j.ins.2018.12.048.

[27] M. chenaghlu, M. Balafar, M. Derakhshi, "A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation," *Signal Processing*, vol. 157, p. 1, 2019, doi: 10.1016/j.sigpro.2018.11.010.

[28] X. Zhang, X. Wang, "Multiple-image Encryption algorithm based on DNA encoding and chaotic system," *Multimedia Tools and Applications*, vol. 78, no. 6, USA, 2019, doi: 10.1007/s11042-018-6496-1.

[29] M. Al-Dwairi, A. Hendi, Z. AlQadi, ''An efficient and highly secure technique to encrypt-decrypt color images,'' *Engineering, Technology & Applied Science Research*, vol. 9, no. 3, pp. 4165-4168, 2019, doi: 10.48084/etasr.2525.

[30] A. Hendi, M. Dwairi, Z. Al-Qadi, M. Soliman, ''A novel simple and highly secure method for data encryption-decryption,'' *International Journal of Communication Networks and Information Security*, vol. 11, no. 1, pp, 232-238, 2019, doi: 10.17762/ijcnis.v11i1.3999.

[31] Z. Alqadi, M. Khrisat, A. Hindi, M. Dwairi, ''Using speech signal histogram to create signal features,'' *International Journal of Engineering Technology Research & Management*, vol. 4, no. 3, pp. 144-153, 2020.

[32] M. Abu-Faraj, Z. Alqadi, K. Aldebei, "Comparative analysis of fingerprint features extraction methods," *Journal of Hunan University Natural Sciences*, vol. 48, no. 12, pp. 177-182, 2021.