


M-folding method-based elliptic curve cryptosystem for industrial cyber-physical system

International Journal of Distributed
Sensor Networks
2019, Vol. 15(10)
© The Author(s) 2019
DOI: 10.1177/1550147719879045
journals.sagepub.com/home/dsn


Taehoon Do¹, Seungwoo Park², Jaehwan Lee² and Sangoh Park²

Abstract

Recently, cyber-physical system is widely used for smart system control in various fields. Various functions of the cyber-physical system must overcome the limited hardware resources constraint of an embedded system. In addition, the data required from the industrial cyber-physical system are critical; therefore, a highly secure encryption technique is required. However, security and computational throughput are incompatible with each other in the cryptographic technique; therefore, the industrial cyber-physical system needs to adopt a highly efficient and secure encryption technique considering the limited available resources. This study applies the m-folding method to the highly secure elliptic curve algorithm to improve efficiency and proposes the cryptosystem optimized for the resource-constrained industrial cyber-physical system. The proposed m-folding method-based elliptic curve encryption showed 50% faster encryption than the existing methods.

Keywords

Cryptography, elliptic curve, cyber-physical system, m-folding, security

Date received: 2 June 2019; accepted: 13 August 2019

Handling Editor: Pascal Lorenz

Introduction

Currently, cyber-physical system¹⁻⁷ that has a wide range of application in industries such as factory, farm, veterinary, transportation, aviation, health care, and chemistry plays an important role in constructing an interaction system between cyber and physical elements. However, several security aspects should be carefully examined to successfully combine cyber-physical systems with various technologies. Current networks of energy grids, water supply, logistics, transportation, and so on comprise structures that are vulnerable to external malicious attacks. Data in the industrial cyber-physical system are usually critical, and they must be protected.

The cyber-physical system is based on an embedded system; therefore, it is necessary to consider the limited system resources in such a system. In addition, the encryption and decryption times are crucial because a

low delay in control and response is required in cyber-physical system. In other words, the functions of the cyber-physical system should be designed based on algorithms with low computational complexity and high efficiency of resources. Generally, when the key length of the asymmetric key system increases, the encryption efficiency decreases due to the increased CPU time or memory requirements while implementing a cryptographic system. The amount of data to be sent and received also increases, thereby requiring more

¹Korea Securities Computer Corporation (KOSCOM), Seoul, Korea

²School of Computer Science and Engineering, Chung-Ang University, Seoul, Korea

Corresponding author:

Sangoh Park, School of Computer Science and Engineering, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 06974, Korea.
Email: sopark@cau.ac.kr



bandwidth for communication.^{8–10} Therefore, existing cryptosystems such as Rivest–Shamir–Adleman (RSA)¹¹ or homomorphic encryption¹² are difficult to deploy in the cyber-physical system environment due to the large key size of the encrypted data.

Elliptic curve cryptosystem^{13–16} is a cryptographic system that uses the computational complexity of the discrete logarithm problem on elliptic curves. It is an asymmetric key-based cryptosystem that has gained popularity in recent years due to the advantage of having a similar degree of safety with shorter key lengths compared to the conventional asymmetric key-based cryptosystem. Especially, security of the elliptic curve cryptosystem increases almost exponentially with the increase in the key length, and it has a considerable advantage in terms of rate of increase in the key length. This is a result of long-term technological advances compared to the existing asymmetric key cryptographic system with a quasi-index function. For example, the RSA cryptosystem must use approximately 15,000 bits of composite number to provide security similar to the elliptic curve cryptosystem of approximately 512 bits.¹¹

This study proposes the m-folding method-based elliptic curve cryptosystem for the industrial cyber-physical system. M-folding improves the scalar multiplication operation in elliptic curve cryptosystem by representing the number of multiplication count of a point as a bit string and further dividing it into arbitrary length. The elliptic curve cryptosystem architecture based on the m-folding method for the industrial cyber-physical system is also proposed in this study. Performance evaluation shows that the proposed method shows 50% faster encryption than the existing methods.

The rest of the article is organized as follows. Section “Related work” describes the related work and section “M-folding method-based elliptic curve encryption technique” describes the elliptic curve-based cryptosystem suitable for the industrial cyber-physical system. The performance evaluation results of the proposed method are presented in section “Performance evaluation.” Finally, section “Conclusion and future work” presents our conclusion and suggests future work.

Related work

Cyber-physical system

The cyber-physical systems^{1–7} are the next-generation network-based distributed control systems that combine physical systems having sensors and actuators with computing elements that control them. The advances in the cyber-physical system technology are a key factor in making life more convenient and efficient; however, the risk and vulnerability in terms of safety have increased. Cyber terrorism is no longer limited to information

security in virtual spaces such as computers or Internet servers but is capable of attacking the actual control system, which is a critical issue that could directly threaten our lives and shake the foundations of nations. An example of the first attack on control systems is the Stuxnet virus¹⁷ discovered in Iran in 2010. Cyberattacks on various daily life or critical infrastructure have been reported since then. Therefore, the study of safety and security of cyber-physical systems has become more urgent and important than anything else.

A low delay in control and sensing is necessary in the cyber-physical system; therefore, the encryption and decryption times are crucial in such systems. The functions of the cyber-physical system should be designed based on algorithms with low computational complexity and memory usage. Symmetric key algorithms must use the same encryption key between the receiver and the sender and deliver the encryption key. It is not applicable for the cyber-physical system because a large number of devices need to communicate in order to share the same encryption/decryption key. Public key systems such as RSA and elliptic curve cryptosystem were designed to overcome the aforementioned limitations. The elliptic curve cryptosystem is more suitable in the cyber-physical system scenario due to its smaller key size. When the key length of the public key system increases, the number of computations and memory usage also increases with the increased size of the encryption data. Therefore, existing cryptosystems such as RSA¹¹ and homomorphic encryption¹² are difficult to deploy in the cyber-physical system environment.

Elliptic curve cryptography is gaining popularity in the field of cyber-physical system security due to its public key scheme and smaller key size compared to existing asymmetric key-based cryptosystem. Elliptic curve cryptography has several applications in the cyber-physical systems such as medical cyber-physical system encryption,¹⁸ wireless sensor network infrastructure,¹⁹ and key agreement in the smart grid.²⁰ However, none of these applications considered improving the encryption and decryption scheme for elliptic curve cryptography deployed in the cyber-physical system environment.

Elliptic curve cryptography

Elliptic curve cryptography^{8,9} is a cubic equation with two variables. The general elliptic curve equation is given by equation (1)

$$y^2 + b_1xy + b_2y = x^3 + a_1x^2 + a_2x + a_3 \quad (1)$$

Elliptic curve in real number space uses an elliptical curve in a special category such as equation (2)

$$y^2 = x^3 + ax + b \quad (2)$$

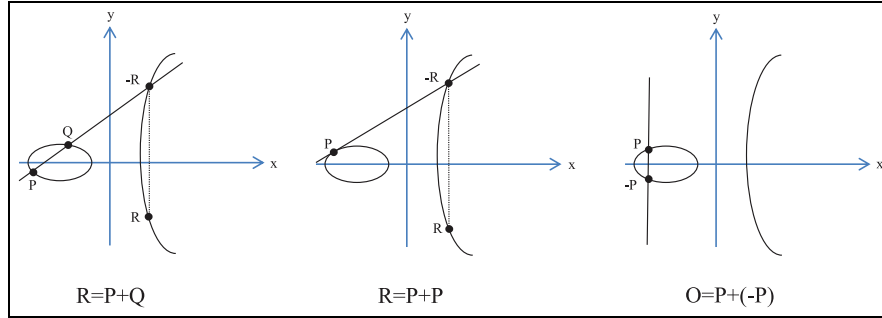


Figure 1. Three cases of elliptic curve.

In an elliptic curve encryption technique, if $4a^3 + 27b^2 \neq 0$, the curve is a nonsingular elliptic curve, or it represents a unique elliptical curve. A regular elliptical curve cannot have three different roots. Figure 1 shows an example of finding another point on a curve when two points on a curve are added in an elliptic curve.

In case of $P \neq Q$

$$\begin{aligned} \lambda &= \frac{y_2 - y_1}{x_2 - x_1} \\ x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1 \end{aligned} \quad (3)$$

In case of $P = Q$

$$\begin{aligned} \lambda &= \frac{3x_1^2 + a}{2y_1} \\ x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1 \end{aligned} \quad (4)$$

In the case when $P = -P$, the straight line connecting these two points does not intersect at the other point of the elliptic curve; therefore, the other point is defined as O , and it becomes an identity element for addition.

Fast scalar multiplications for elliptic curve cryptography

The main computation performed by the elliptic curve-based encryption technique is the computation: $d * P$ that multiplies the point P by d . In other words, point P is added d times. As the addition in elliptic curve shows, multiplication, addition, and inverse calculations must be performed. Furthermore, scalar multiplication that can be performed by repetition of the aforementioned computations requires considerable computational time, which is an important factor determining the performance of the entire cryptosystem. Research works^{21–28} have been conducted on algorithms that can speed up the d iterative addition operations of point P in large numbers that usually exceed 160 bits.

```

1:  $R \leftarrow P$ 
2:  $L \leftarrow$  the number of figures of  $d$ 
3: for  $i \leftarrow L - 2$  to 0 do
4:    $R \leftarrow 2R$ 
5:   if  $d_i = 1$  then
6:      $R \leftarrow P + R$ 
7:    $i \leftarrow i - 1$ 
8: return  $R$ 

```

Figure 2. Algorithm of binary method.

Therefore, to achieve high efficiency of the cyber-physical system with limited resources, the performance of scalar multiplication should be improved. First, we discuss the existing scalar multiplication algorithm and then propose a new algorithm that performs better than the existing one.

Binary method. The binary method^{23,24} is a technique of expressing d in binary numbers by reading from the most significant bits sequentially and processing according to the following conditions:

If the bit is 0, perform point doubling.

If the bit is 1, perform point addition with P after point doubling.

The binary method represented by a specific algorithm is shown in Figure 2. For example, as shown in Figure 3, to calculate $186P$, 186 is expressed as binary number 10111010 and processed according to the algorithm in Figure 2. However, simply adding Pd times increases the number of addition operations and computation time.

M-ary method. The m-ary method²⁵ is a computation technique for reducing the number of digits by generalizing what is expressed as binary number in the binary method. The format of expression used by this method

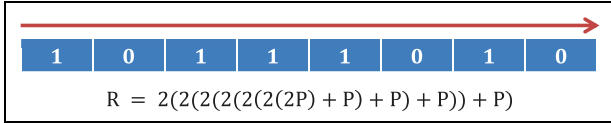


Figure 3. Computation process of binary method.

```

1:  $P_0 \leftarrow O$  (point of infinity)
2:  $L \leftarrow$  the number of figures of  $d$ 
3: for  $i \leftarrow 1$  to  $m - 1$  do
4:    $P_0 \leftarrow P_{i-1} + P$ 
5:    $i \leftarrow i + 1$ 
6:  $R \leftarrow P_{d_{L-1}}$ 
7: for  $i \leftarrow L - 2$  to  $0$  do
8:    $R \leftarrow mR$ 
9:    $R \leftarrow R + P_{d_i}$ 
10:   $i \leftarrow i - 1$ 
11: return  $R$ 

```

Figure 4. Algorithm of m-ary method.

is $(d_{L-1}, d_{L-2}, d_{L-3}, \dots, d_0)$, which is faster by a small degree than the binary method. In the m-ary method, d is expressed as an m-ary number having L figures. iP ($i = 1, 2, \dots, m - 1$) is calculated in advance, and then the value of dP is calculated by performing point addition m times and iP times from d_{L-1} to d_0 , respectively.

The m-ary method represented by a specific algorithm is shown in Figure 4. For example, to calculate $186P$, 186 is expressed as tetramal 2322 and processed according to Figure 5. Storing even numbers such as $2P$, $4P$, and $6P$ in the m-ary method can waste memory. For example, $2P$ is two times P , $4P$ is four times P , and $6P$ is two times $3P$. As shown by the algorithm in Figure 4, point doubling is performed on each loop basically and with only P or $3P$. $2P$, $4P$, and $6P$ can be calculated with the same number of calculations. Therefore, information from $2P$, $4P$, and $6P$ being unnecessary information can waste memory.

Window method. In the window method,²⁶ when d is expressed in binary form, a set of bits less than a certain length, w , with start and end as 1 is called windows. It is calculated in advance and processed in a manner similar to the m-ary method. The window method is represented by a specific algorithm shown in Figure 6. For example, to calculate $186P$, 186 is expressed as binary 10111010 as shown in Figure 7 and processed according to the above algorithm.

The window method stores only odd multiples such as P , $3P$, and $5P$; therefore, it does not encounter the memory space problem of the m-ary method.

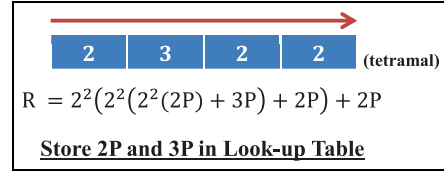


Figure 5. Computation process of m-ary method.

```

1:  $P_0 \leftarrow P$ 
2:  $T \leftarrow 2P$ 
3:  $L \leftarrow$  the number of figures of  $d$ 
4: for  $i \leftarrow 1$  to  $2^{w-1} - 1$  do
5:    $P_i \leftarrow P_{i-1} + T$ 
6:    $i \leftarrow i + 1$ 
7:  $R \leftarrow O$  (point of infinity)
8:  $i \leftarrow L - 1$ 
9: while  $i \geq 0$  do
10:  if  $d_i = 1$  then
11:     $j \leftarrow i - w + 1$ 
12:    if  $j \leq 0$  then
13:       $j \leftarrow 0$ 
14:    while  $d_i = 0$  do
15:       $j \leftarrow j + 1$ 
16:     $tmp \leftarrow (d_i, \dots, d_j)_2$ 
17:    for  $k \leftarrow j$  to  $i$  do
18:       $R \leftarrow 2R$ 
19:       $R \leftarrow R + P_{tmp-1/2}$ 
20:       $i \leftarrow j - 1$ 
21:    else
22:       $R \leftarrow 2R$ 
23:       $i \leftarrow i - 1$ 
24: return  $R$ 

```

Figure 6. Algorithm of window method.

Consequently, it is possible to store a higher multiple of P values using the same memory space when compared to the m-ary method. However, the memory may be wasted in other ways, for example, if the window size is three, the look-up table stores $3P$, $5P$, and $7P$; however, there may actually exist values referenced less than two times. In addition, as shown in Figure 7, the value $7P$ is not used, which can result in memory wastage.

Scalar-folding method. In the scalar-folding method,²⁷ d is expressed as a binary number. Furthermore, its bit string $(d_{L-1}, d_{L-2}, d_{L-3}, \dots, d_0)$ is divided into $(d_{L-1}, \dots, d_{L/2})$ and $(d_{L/2-1}, \dots, d_0)$. If L is an odd number, only the upper bits except the lower 1 bit are considered and the lower 1 bit is calculated at the end. The value of the first start bit of the upper part, $2^{L/2}$, is calculated and stored in the look-up table. Furthermore, to calculate the upper and lower parts at

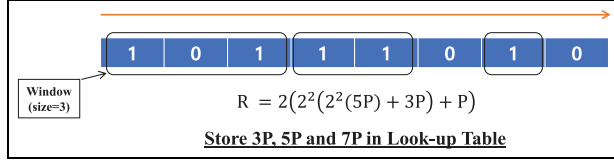


Figure 7. Computation process of window method.

```

1:  $P_0 \leftarrow P$ 
2:  $P_1 \leftarrow P$ 
3:  $L \leftarrow$  the number of figures of  $d$ 
4: for  $i \leftarrow K$  to  $L/2$  do
5:    $P_1 \leftarrow 2P_1$ 
6:    $i \leftarrow i + 1$ 
7:  $K \leftarrow L - 1$ 
8:  $R \leftarrow O$  (point of infinity)
9: for  $i \leftarrow K$  to  $1/2$  do
10:  if  $d_i = 1$  then
11:     $R = R + P_1$ 
12:  if  $d_{i-L/2} = 1$  then
13:     $R = R + P_0$ 
14:     $R = 2R$ 
15:     $i \leftarrow i - 1$ 
16: if  $(L \bmod 2) \neq 0$  then
17:    $R = 2R$ 
18:   if  $d_0 = 1$  then
19:     $R = R + P_0$ 
20: return  $R$ 

```

Figure 8. Algorithm of scalar-folding method.

the same time, the value of dP is calculated using $\sum_{i=0}^{(L/2)-1} 2^i((d_{i+L/2}2^{L/2}P) + d_iP)$. For example, to calculate $186P$, 186 is expressed as binary 10111010 and processed according to the algorithm, as shown in Figure 8.

Analysis of existing methods. The existing methods solely focus on decreasing the number of computations in a general scenario. In the cyber-physical system environment, where a large number of devices are deployed and authentication keys do not change frequently with p , a scheme that can take advantage of pre-computed look-up table can greatly improve the real-time encryption/decryption performance of the elliptic curve-based cryptosystem. We propose the computation method that can flexibly fold a bit string in arbitrary size and utilize a pre-computed look-up table.

M-folding method-based elliptic curve encryption technique

M-folding method

This study proposes the m-folding method. It divides a bit string into m parts and calculates all the parts

```

1:  $P_0 \leftarrow P$ 
2:  $L \leftarrow$  the number of figures of  $d$ 
3:  $S \leftarrow (L - 1 \bmod m)$ 
4: for  $i \leftarrow 1$  to  $m - 1$  do
5:    $P_i \leftarrow P_{i-1}$ 
6:   for  $j \leftarrow 0$  to  $S - 1$  do
7:      $P_i = 2P_i$ 
8:      $j \leftarrow j + 1$ 
9:    $i \leftarrow i + 1$ 
10:  $R \leftarrow P_{m-1}$ 
11: for  $i \leftarrow 0$  to  $m - 1$  do
12:  if  $d_{D_i} = 1$  then
13:     $R = R + P_i$ 
14:     $i \leftarrow i + 1$ 
15: for  $i \leftarrow 1$  to  $S$  do
16:    $R \leftarrow 2R$ 
17:   for  $j \leftarrow 0$  to  $m - 1$  do
18:      $D_j \leftarrow D_j - 1$ 
19:      $j \leftarrow j + 1$ 
20:   for  $j \leftarrow 0$  to  $m - 1$  do
21:    if  $d_{D_j} = 1$  then
22:       $R = R + P_j$ 
23:       $j \leftarrow j + 1$ 
24:    $i \leftarrow i + 1$ 
25: return  $R$ 

```

Figure 9. Algorithm of proposed method.

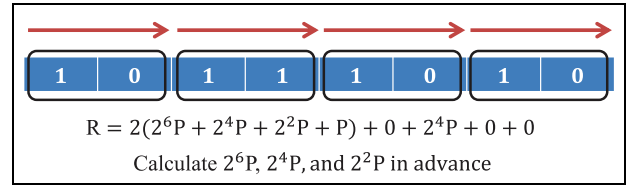


Figure 10. Computation process of proposed method.

simultaneously. It is an extension of the scalar-folding method that divides the existing bit string into two parts. First, d is expressed as binary number and its bit string $(d_{L-1}, d_{L-2}, d_{L-3}, \dots, d_0)$ is divided into $(d_{L-1}, \dots, d_{L-1-L/m})$, $(d_{L-2-L/m}, \dots, d_{L-2-2L/m})$, and so on. If L is not divisible by m , the lower part $(L \bmod m)$ is calculated later, and the starting bit of each part is calculated and stored in the look-up table. The value of dP is then calculated using $\sum_{i=0}^{L/2-1} 2^i((d_{i+L/2}2^{L/2}P) + d_iP)$ to calculate the upper and lower parts at the same time. For example, to calculate $186P$, 186 is expressed as binary 10111010 as shown in Figure 9 and processed according to Figure 10.

Elliptic curve-based cryptosystem architecture

This study designed the m-folding method-based elliptic curve cryptosystem architecture for the industrial cyber-physical system. Figure 11 shows the proposed

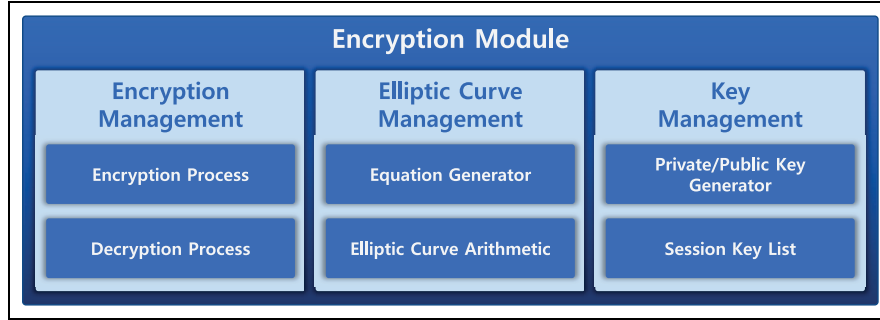


Figure 11. Elliptic curve-based cryptosystem architecture.

architecture. M-folding method-based elliptic curve cryptosystem comprises an encryption management module that performs encryption and decryption, an elliptic curve management module that executes an m-folding method-based elliptic curve encryption algorithm, and a key management module that manages the encryption keys.

Encryption management. This module encrypts the messages generated by the message generator in the message module through the encryption process or receives the encrypted messages from the network and sends the messages decrypted by the decryption process to the message parser in the message module. The encryption and decryption algorithm simulates the ElGamal cryptosystem among asymmetric key-based encryption techniques. The encryption and decryption processes are given by equations (5) and (6). First, a point P corresponding to the plain text is selected on the curve. Furthermore, two points to be used as ciphertexts are calculated using equations (5) and (6) where r is an arbitrary integer; e_1 and e_2 are recipient's public key to receive the encrypted message at the point on the elliptic curve

$$C_1 = r \times e_1 \quad (5)$$

$$C_2 = P + r \times e_2 \quad (6)$$

The decoding calculates P using equation (7), where d is any integer

$$P = C_2 - (d \times C_1) \quad (7)$$

Stability of the elliptic curve²⁸ is based on the computational complexity of solving the elliptic curve logarithm problem. For example, if an intruder has obtained C_1 and e_1 over the network, they can find r using equation $C_1 = r \times e_1$. However, the intruder cannot calculate plain text P in $C_2 = P + r \times e_2$ easily because it is very difficult to calculate r as an elliptic curve logarithm problem.

Elliptic curve management. This study uses the elliptic curve equation recommended by the Korean certificate-based digital signature algorithm using elliptic curve (EC_KCDSA).⁴ The equation generator generates and manages finite field variables that are necessary to define the elliptic curve equation. EC-KCDSA recommends two types of elliptic curves to be used for encryption: arbitrary curve given by equation (8) and the Koblitz curve given by equation (9)

$$y^2 = x^3 - 3x + b \quad (8)$$

$$y^2 = x^3 + b \quad (9)$$

The elliptic curve equation in this study is based on the Koblitz elliptic curve equation because it can be calculated efficiently and can consequently ensure faster encryption. The elliptic curve arithmetic module defines the operations required for the encryption module, for example, finding points on an elliptic curve or addition operation between two points.

Key management. The private/public key generator generates public and private keys for encryption and decryption based on elliptic curve. The public key selects an arbitrary point e_1 on an elliptic curve equation and an elliptic curve. Then, an arbitrary integer d is selected and e_2 is calculated as $e_2 = d \times e_1$. Here, e_1 , e_2 , and the elliptic curve are declared as its own public key, and d being a private key is not disclosed. The session key list module manages the sessions between nodes, stores the keys used for each session in a table, and manages the sessions and related keys through key distribution and deletion.

Performance evaluation

During performance evaluation, performance of the m-folding method was compared with existing scalar multiplication algorithms in the following three conditions: (1) when look-up tables are not used, (2) when look-up tables are used, and (3) when pre-computed look-up

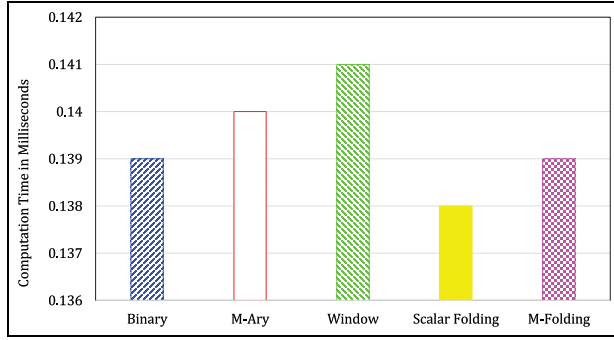


Figure 12. Case when look-up table is not used.

tables are used. When the look-up table is calculated in advance, it is not created each time the algorithm is executed; however, the created table is stored at a known location and reused. The performance evaluation compares the processing speed of $d * P$ using a constant d that is 160 bits long and point P on an elliptic curve. While using look-up tables, cases such as m-ary, window, folding, and m-folding need to be considered due to the characteristics of the algorithm; however, scalar-folding is excluded from the performance comparison because scalar-folding is the case when the value m is 2 in m-folding.

Case when look-up table is not used

Figure 12 shows the performance result without the look-up table. Each algorithm showed a processing speed between 138 and 141 ms, thereby showing no significant difference in performance.

Case when look-up table is used

Figure 13 shows the performance when the look-up table is used. The window technique showed better performance when the look-up table had larger memory. As the size of the look-up table increases, the performance of m-ary method increases. However, after a certain level, the performance decreases because when a binary number is converted to the given base, the larger the value of m , the larger are the required memory and conversion time. In the proposed m-folding technique, the values stored in the look-up tables are $P * 2^n$ that do not affect the performance significantly because they are necessary for the scalar multiplication.

Case when look-up table is created in advance and recycled

The performance results in Figure 14 show that when the look-up table is pre-computed, the processing speed increases when the additional storage size increases

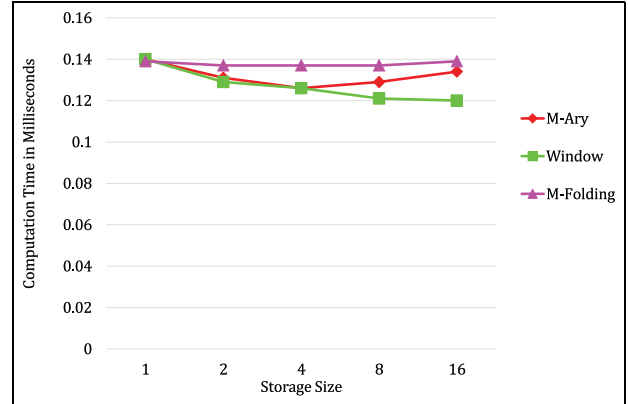


Figure 13. Case when look-up table is used.

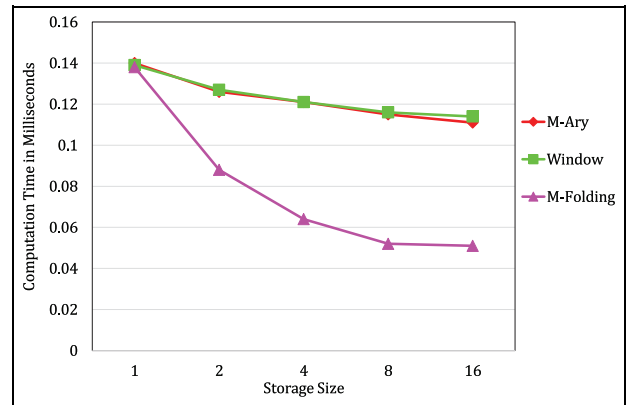


Figure 14. Case when look-up table is pre-computed and recycled.

because the time to re-create the tables is saved. Compared to the window method, the m-ary method showed slightly better performance. M-folding method showed better performance than other algorithms when the storage space was two or more, and the processing speed was approximately two times faster when the storage space was four or more. On the contrary, the rate of improvement of the processing speed relative to storage size gradually decreased because the bit string of d is divided into a number 2^m ; therefore, as the value of m increases, the rate of increase in value $2^{L-1-L/m} * P$ stored in the look-up table exponentially decreases.

As a result, the proposed m-folding method shows good performance while using a pre-stored look-up table. Using pre-stored look-up table is suitable in an environment where the value of p does not change much. The cryptosystem used in this study does not change the value of p ; therefore, the authentication key of other controllers can be used for the look-up table in advance by the proposed m-folding method to be used later.

Conclusion and future work

This study analyzed the elliptic curve algorithm that is more efficient than the asymmetric-based encryption technique and designed the encryption technique with improved performance by applying the m-folding method that is more efficient than the existing scalar multiplication algorithm. This technique can mitigate the resource constraint problem; therefore, we used it to design the m-folding method-based elliptic curve cryptosystem structure suitable for the industrial cyber-physical system. However, the performance evaluation results showed excellent performance only when the look-up table is pre-computed and recycled.

In the future, it is necessary to research a cryptosystem that can improve the security of the industrial cyber-physical system using an encryption technique with high security and efficiency such as elliptic curve algorithm or comprehensively researching a high-performance technique regardless of a look-up table.


Declaration of conflicting interests


The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was carried out with the support of the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2019-2018-0-01799) supervised by the Institute for Information & Communications Technology Planning & Evaluation (IITP) and the Chung-Ang University Graduate Research Scholarship in 2018.

ORCID iDs

Seungwoo Park  <https://orcid.org/0000-0001-7480-1589>

Sangoh Park  <https://orcid.org/0000-0002-1832-3532>

References

- Lee EA. Cyber physical systems: design challenges. In: *Proceedings of the 2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC)*, Orlando, FL, 5–7 May 2008, pp.363–369. New York: IEEE.
- Park SO, Do TH, Jeong YS, et al. A dynamic control middleware for cyber physical systems on an IPv6-based global network. *Int J Commun Syst* 2013; 26(6): 690–704.
- Lee J, Bagheri B and Kao H. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manuf Lett* 2015; 3: 18–23.
- Lee EA and Seshia SA. *Introduction to embedded systems: a cyber-physical systems approach*. Cambridge, MA: MIT Press, 2016.
- Elshenawy M, Abdulhai B and El-Dariby M. Towards a service-oriented cyber-physical systems of systems for smart city mobility applications. *Future Gener Comp Sys* 2018; 79: 575–587.
- Serpanos D. The cyber-physical systems revolution. *Computer* 2018; 51(3): 70–78.
- Lee HS, Lee JH, Nam SH, et al. Efficient heterogeneous network-routing method based on dynamic control middleware for cyber-physical system. *J Sensors* 2018; 2018: 3176967.
- Rivest RL, Shamir A and Adleman L. A method for obtaining digital signatures and public-key cryptosystems. *Commun ACM* 1978; 21(2): 120–126.
- Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE T Inform Theory* 1985; 31(4): 469–472.
- Kumar Y, Munjal R and Sharma H. Comparison of symmetric and asymmetric cryptography with existing vulnerabilities and countermeasures. *Int J Comput Sci Manag Stud* 2011; 11(3): 60–63.
- Lee Y and Haas ZJ. Authentication in very large ad hoc networks using randomized groups. In: *Proceedings of the 2005 IEEE 16th international symposium on personal, indoor and mobile radio communications*, Berlin, 11–14 September 2005, vol. 3, pp.1989–1993. New York: IEEE.
- Wang L, Li J and Ahmad H. Challenges of fully homomorphic encryptions for the internet of things. *IEICE T Inf Syst* 2016; 99(8): 1982–1990.
- Koblitz N. Elliptic curve cryptosystems. *Math Comput* 1987; 48(177): 203–209.
- Renes J, Costello C and Batina L. Complete addition formulas for prime order elliptic curves. In: *Proceedings of the annual international conference on the theory and applications of cryptographic techniques*, Vienna, 8–12 May 2016, pp.403–428. Berlin: Springer.
- Semaev I. New algorithm for the discrete logarithm problem on elliptic curves, 2015, <https://arxiv.org/abs/1504.01175>
- Godor G and Imre S. Elliptic curve cryptography based authentication protocol for low-cost RFID tags. In: *Proceedings of the 2011 IEEE international conference on RFID-technologies and applications*, Sitges, 15–16 September 2011, pp.386–393. New York: IEEE.
- Farwell P and Rohozinski R. Stuxnet and the future of cyber war. *Survival* 2011; 53(1): 23–40.
- Kocabas O, Soyata T and Aktas K. Emerging security mechanisms for medical cyber physical systems. *IEEE-ACM T Comput Bi* 2016; 13(3): 401–416.
- Barbareschi M, Battista E and Casola V. On the adoption of FPGA for protecting cyber physical infrastructures. In: *Proceedings of the 2013 eighth international conference on P2P, parallel, grid, cloud and internet computing*, Compiegne, 28–30 October 2013, pp.430–435. New York: IEEE.
- Wu F, Xu L, Li X, et al. A lightweight and provably secure key agreement system for a smart grid with elliptic curve cryptography. *IEEE Syst J* 2018; 13: 2830–2838.

21. Ansari B and Hasan MA. High-performance architecture of elliptic curve scalar multiplication. *IEEE T Comput* 2008; 57(11): 1443–1453.
22. Rivain M. Fast and regular algorithms for scalar multiplication over elliptic curves. *IACR Cryptol ePrint Arch* 2011; 2011: 338.
23. Rashidi B, Sayedi SM and Farashahi RR. High-speed hardware architecture of scalar multiplication for binary elliptic curve cryptosystems. *Microelectron J* 2016; 52: 49–65.
24. Pontie S, Maistri P and Leveugle R. Dummy operations in scalar multiplication over elliptic curves: a tradeoff between security and performance. *Microprocess Microsy* 2016; 47: 23–36.
25. Azarderakhsh R and Karabina K. Efficient algorithms and architectures for double point multiplication on elliptic curves. In: *Proceedings of the third workshop on cryptography and security in computing systems*, Prague, 20–20 January 2016, pp.25–30. New York: ACM.
26. Rasmi M, Sokhon AA, Daoud MS, et al. A survey on single scalar point multiplication algorithms for elliptic curves over prime fields. *IOSR J Comput Eng* 2016; 18: 31–47.
27. Wu K, Li H and Zhu D. Fast and scalable parallel processing of scalar multiplication in elliptic curve cryptosystems. *Secur Commun Netw* 2012; 5(6): 648–657.
28. Langley A, Hamburg M and Turner S. Elliptic curves for security. RFC no. 7748, January 2016, <http://www.rfc-editor.org/info/rfc7748>