Pol. J. Appl. Sci., **2018**, *4*, 102-109

ANALYSIS OF ELLIPTIC CURVES FOR CRYPTOGRAPHIC APPLICATIONS DEDICATED TO DEVICES WITH LIMITED COMPUTATIONAL RESOURCES

Wiesław Maleszewski

Faculty of Computer Science and Food Science
Lomza State University of Applied Sciences, Lomza, Poland

E-mail: wmaleszewski@pwsip.edu.pl

Abstract: The dynamic development of the Internet of Things infrastructure contributes to the need for a new look at the methods of protecting information transmission. Most modern systems were designed for devices with easy access to power. Very often the Internet of Things devices have both limited computing power and very limited energy resources. These conditions contribute to the need to optimize the computing costs of algorithms which ensure communication protection. The purpose of the following work is to provide a concise introduction of the properties of certain algebraic groups generated by the most popular elliptic curve subfamilies and a comparison of the computational costs of the arithmetic operations of adding and doubling points in these groups.

Key words: Cryptography, RSA, ECC, IoT

DOI: 10.34668/PJAS.2018.4.3.04

Introduction

Nowadays, the development of a new infrastructure has changed all areas of our functioning – which is the Internet of Things. Its market in the first two quarters of 2018 recorded a dynamic increase, exceeding in total 7 billion IoT devices in use [1].

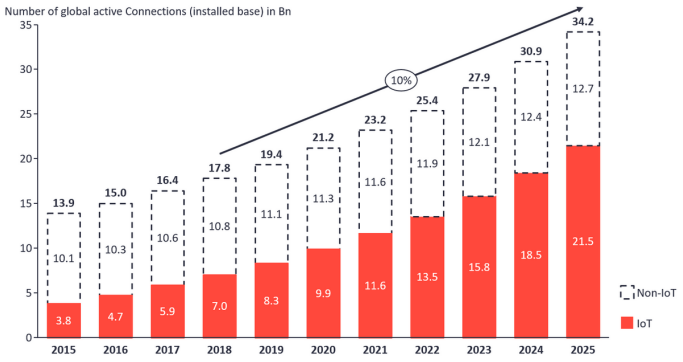

Fig. 1: Total number of active device connections worldwide [1].

According to the latest International Data Corporation (IDC) reports, the Internet of Things market is growing at a rate of around 20% yearly. And in 2020 its value is estimated to exceed one trillion US dollars. In 2021, 16 out of 28 billion devices connected to the Internet will be IoT devices.

On the basis of the recent investor's presentation, global data traffic is expected to grow at 45%. IoT spending in Central and Eastern Europe will record a five-year compound annual growth rate (CAGR) of 18.1% between 2017 – 2022 with expenditures surpassing 22 billion USD in 2022 [1].

Due to the rapid development of the Internet of Things technology developers of modern devices point to its new possibilities of innovative use in various areas.

In the article [2], the authors present the most important areas of applications of the Internet of Things, among which there are such areas as: construction, healthcare, industry and production, transport, public safety and IT systems.

In addition to many positive aspects, there are also threats. Rapidly developing technology requires us to look for better protection of our data – which is transmitted more often on the Internet and becomes more threatened by various attacks.

Frost and Sullivan's recently analyzed Global Industrial Cybersecurity Services Market. Companies that are eager to grow within the industrial cybersecurity market can find opportunities through [3]:

- Providing integrated platforms that can deploy a range of services to enhance the security posture of end users while incorporating the best security practices.
- Using automated management services and advanced analytics to develop a comprehensive service portfolio that can be adapted for all types of end users.
- Offering flexible pricing models, such as Cybersecurity as a Service (CSaaS), and lifetime services to increase accessibility across industries at a lower cost.

Description of the problem

A serious problem of security threats to IoT devices is a consequence of the fact that many of these IoT devices have severe operational limitations on their physical size and by extension the computational power available to

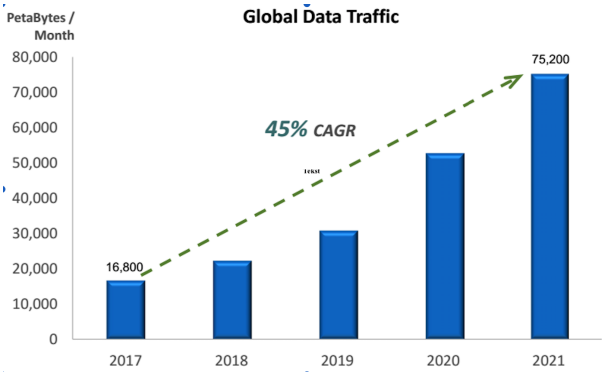

Fig. 2: Total number of active device connections worldwide [1].

them. These constraints often make them unable to directly use basic security measures such as implementing firewalls or using strong cryptosystems to encrypt their communications with other devices [4].

The IoT world is based on microcontrollers adapted for direct cooperation with various external devices, including those to which the traditional microprocessor would require the use of additional peripherals. (As Wikipedia says a microcontroller is a small computer on a single integrated circuit which contains one or more CPUs along with memory and programmable peripherals [5].)

At IBM's Think 2018 conference, the company announced the creation of the world's smallest computer that despite its size and cost (each will cost less than 10 cents to make) can monitor, analyze, communicate and even act on data. Each computer can hold as many as one million transistors, while network communication is handled by LEDs, and a solar cell provides power [6].

Current trends are heading towards the production of devices with the smallest dimensions, which exclude the use of high-capacity batteries.

The trend of the constructors of electronic devices is to minimize energy consumption in order to maximize the working time of the mobile device after charging or changing the battery.

Energy efficiency usually requires the microcontroller mode of the core and as many of the peripheral systems as possible, to be in sleep mode and at the same time reduced energy consumption and are only awoken for the time of important tasks. In most modern microcontrollers, peripheral modules, such as communication interfaces and DMA controllers, work independently of the CPU and are able to perform certain tasks independently without switching them on. From the point of view of energy efficiency, it is very important that they are carried out in the shortest possible time. Then the average supply current consumed by the device is little, which contributes to a longer battery life.

Motivation and methodology

Memory size, power consumption and computational performance are the most important features of smart IoT designers that use complex algorithms. The present solutions available usually belong to one of two groups: they have low power consumption but limited computational efficiency and memory size or are characterized by higher power consumption, offering in return more efficient processors and more memory.

Securing millions of non-standard devices such as home thermometers, smart TVs and cars is not a trivial task. It is somewhat simpler if they have easy stripped down embedded processors, but often they contain full-fledged and powerful network-connected operating systems, with all the security problems those present [3].

Related Work

Modern algorithms are based on so-called computationally difficult problems. One of the problems computationally difficult is the problem of factorization grounded on the asymmetric RSA algorithm designed by Ron Rivest, Adi Shamir and Leonard Adleman in 1977. Another computationally difficult problems is the problem of discrete logar on which base Elliptic Curve Cryptography (ECC). These problems do not guarantee security against attacks from quantum computers, and the costs of encrypting and decrypting data, especially in RSA, are so high that it becomes difficult to apply this algorithm to devices with limited computing resources.

RSA

The RSA algorithm is currently one of the most popular cryptographic algorithms that can be used for both message encryption and digital signatures. It was designed by Ron Rivest, Adi Shamir and Leonard Adleman in 1977, and through the use of two related public and private keys gave the new possibility of secure communication. The algorithm solves the problem of key distribution characteristic of symmetric algorithms because the public key is explicit assumed by future correspondence recipients in the broadcast mode, so that everyone with its use could execute the ciphertext of the sent message. Decrypting the ciphertext requires a private key whose only possessor should be the recipient of the correspondence. The security of RSA encryption is based on issues of large complex numbers factorization. Despite the growing computational resources of modern computers, it still gives very high security guarantees. A potential threat to the functioning of this system is the development of quantum computing; however, in this article these issues will not be considered; interested readers should refer to [7, 8].

Generating keys in RSA

The procedure for generating keys in the RSA algorithm is carried out according to the following scheme:

1. The choice of two different prime numbers p, q, these numbers should be chosen at random and consist of a similar number of bits.
2. The calculation of the product $n = p \cdot q$, whose length is treated as the length of the RSA key.
3. The calculation for n values of the Euler function [9], using the following form:

$$\varphi(n) = \varphi(p) \cdot \varphi(q) = (p-1) \cdot (q-1) \tag{1}$$

4. Reconciliation of an integer e, from 1to $\varphi(n)$ so that the numbers e and $\varphi(n)$ are relatively prime [Number e is used as the exponent of the public key]
5. Determining the number d that fulfills the condition: $d \cdot e = 1 \mod \varphi(n)$. The number d is used as an exponent of the private key.
6. The choice of two different prime numbers p, q, these numbers should be chosen at random and consist of a similar number of bits.
7. The calculation of the product $n = p \cdot q$, whose length is treated as the length of the RSA key.

The public key consists of the module n and the public exponent e, while the private key consists of the same module n and the private exponent d [7].

Many users can share the value of e. It is recommended that its length should be relatively short, due to the fact that its complexity is significantly dependent on the complexity of calculations during message encryption. The often accepted value of the number e is the first number $2^{16}+1$ (that is 65537). It is also possible to use much more smaller numbers (for example, 3), but under certain circumstances, this weakens the quality of the security.

Encryption is done using the public key (n, e). The message must be divided into parts, then each part should be changed to a number (which must be greater than 0 and less than n). In practice, the message is divided into fragments, each of which consists of a certain number of bits. Then every number included in the message is raised to the power of modulo n :

$$c_i = m_i^e \mod (n) \tag{2}$$

The RSA algorithm can be used repeatedly (using different keys) to encrypt a given message, and then decrypt it in any order. The result will always be the same, regardless of the order of operations. However, you should not encrypt messages in this way more than twice, because then susceptibility to attacks grounded on the Chinese Remainder Theorem has been revealed [10, 11].

Encryption can also be carried out using a private key. The entire procedure is identical to the one described above, with the difference that you will need to use the private key (n, d) for encryption. However, the recipient of the message will have to use the corresponding public key to decrypt the message [10].

Decryption is done using the private key (n, d) The cipher consists of consecutive numbers, less than n. Each number included in the ciphertext is raised to a power equal to d modulo n :

$$m_i = c_i^d \mod n \tag{3}$$

The received numbers of plaintext must be connected in the correct order to create the original message. If the message is encrypted with a private key, you will need to use the corresponding public key to decrypt the message. The decryption procedure is identical to the one described above, with the difference that the public key (n, e) is used.

Elliptic Curve Cryptography

Elliptic Curve Cryptography (ECC),which is a group of asymmetric cryptography techniques based on the arithmetic function of elliptical curves in finite bodies approach [12–14], was introduced independently by two researchers, Neal Koblitz and Victor S. Miller in 1985.

An elliptic curve E over a field F can be given by the Weierstrass equation:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \tag{4}$$

where the coefficients $a_i \in F$. The canonical short Weierstrass form of an elliptic curve is given by the equation:

$$y^2 = x^3 + ax + b, \tag{5}$$

together with a point at infinity $\mathcal{O}$ where the constants a, b meet the additional condition:

$$4a^3 + 27b^2 \neq 0. \tag{6}$$

The algorithm of adding points on the elliptic curve

Let E be an elliptic curve, and $M_1, M_2 \in E$, where $M_1 = (x_1, y_1)$, $M_2 = (x_2, y_2)$, $M_3 = (x_3, y_3)$ and $M_3 = M_1 + M_2$, [15, 16] then:

$$\begin{cases} x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}, \tag{7}$$

where:

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & if \quad (x_1, y_1) \neq (x_2, \pm y_2) \\ \dfrac{3x_1^2 + a}{2y_1} & if \quad (x_1, y_1) = (x_2, \pm y_2) \end{cases}. \tag{8}$$

Remark It is easy to show that along with the aforementioned operation of "addition", and "a point at infinity", elliptic curve E is an Abelian group.

Security ECC is based on the computational complexity of discrete logarithms on elliptic curves – Elliptic Curve Discrete Logarithm Problem (ECDLP).

There is also an ECDSA algorithm for digital signature based on ECC [17].

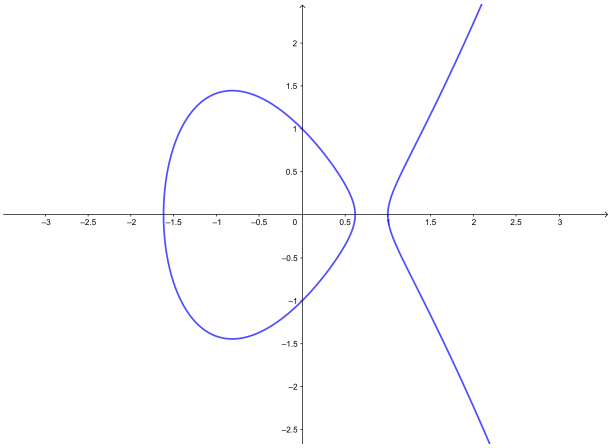

Fig. 3: Elliptic curve $y^2 = x^3 + 5x + 1$

Introduction to Edwards curves

Consider classical circle group:

$$x^2 + y^2 = 1 \tag{9}$$

and two points $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$ on it. Let α_1 and α_2 be the angles such that:

$$P_1 = (x_1, y_1) = (\sin\alpha_1, \cos\alpha_1), \tag{10}$$

$$P_2 = (x_2, y_2) = (\sin\alpha_2, \cos\alpha_2). \tag{11}$$

The sum $P_3 = P_1 + P_2$ is a point on the circle with coordinates (x_3, y_3), where:

$$\begin{aligned} x_3 &= \sin(\alpha_1 + \alpha_2) = \\ &= \sin\alpha_1\cos\alpha_2 + \sin\alpha_2\cos\alpha_1 = x_1y_2 + x_2y_1, \end{aligned} \tag{12}$$

$$\begin{aligned} y_3 &= \cos(\alpha_1 + \alpha_2) = \\ &= \cos\alpha_1\cos\alpha_2 - \sin\alpha_1\sin\alpha_2 = y_1y_2 + x_1x_2. \end{aligned} \tag{13}$$

Therefore the addition formula for points on the circle of radius 1 is [18]:

$$(x_1, y_1) + (x_2, y_2) = (x_1y_2 + x_2y_1, y_1y_2 + x_1x_2) \tag{14}$$

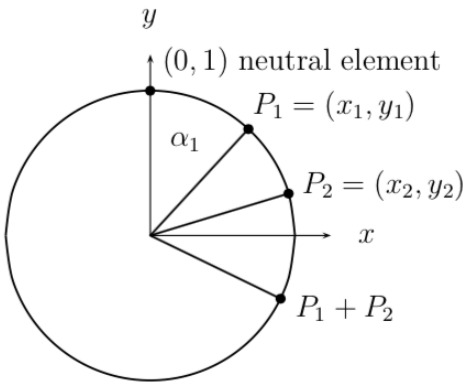

Fig. 4: Addition in classical circle group [18].

Edwards curves

Definition. *Let K be a field with $char(K) \neq 2$. Then an Edwards curve E over K is a curve:*

$$x^2 + y^2 = 1 + dx^2y^2, \tag{15}$$

where $d \in K \setminus \{0, 1\}$.

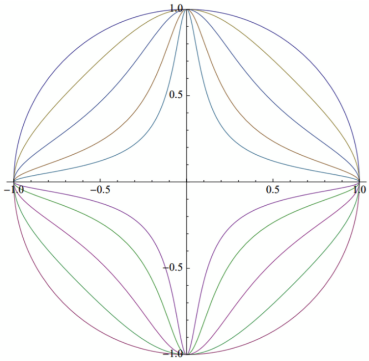

Fig. 5: Edwards curves for $d \in \{0, -2, -10, -50, -200\}$ [19].

Edwards addition law

Let E be an Edwards curve over a finite field K and $char(K) \neq 2$. Let $M_1 = (x_1, y_1)$ and $M_2 = (x_2, y_2)$ be points on E. We then define $M_3 = M_1 + M_2$ as [20]:

$$M_3 = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - x_1x_2}{1 - dx_1x_2y_1y_2}\right), \tag{16}$$

and similarly define $M_4 = 2M_1$ as:

$$M_4 = \left(\frac{2x_1y_1}{1 + dx_1^2y_1^2}, \frac{y_1^2 - x_1^2}{1 - dx_1^2y_1^2}\right). \tag{17}$$

This addition law was proven correct in [21].

Result 1. The zero element of the Edwards addition law is $(0;1)$.

Proof Let $M = (x, y)$ and $\mathcal{O} = (0, 1)$. Then, from the addition law,

$$M + \mathcal{O} = (x, y) + (0, 1) = \\ = \left(\frac{x+0}{1+d\cdot 0}, \frac{y-0}{1-d\cdot 0}\right) = \\ = (x, y) = M \quad (18)$$

Result The inverse of any point (x_1, y_1) is $(-x_1, y_1)$.

Twisted Edwards curves

Bernstein, et. al. in [21] introduced twisted Edwards curves which are curves of the form:

$$ax^2 + y^2 = 1 + dx^2y^2, \quad (19)$$

where $a, d \in K$ are distinct and nonzero [22, 23]. The point $M_3 = M_1 + M_2$ can be determined using the following formula [24]:

$$M_3 = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2}\right), \quad (20)$$

and similarly:

$$M_4 = 2M_1 = \left(\frac{2x_1y_1}{1 + dx_1^2y_1^2}, \frac{y_1^2 - ax_1^2}{1 - dx_1^2y_1^2}\right). \quad (21)$$

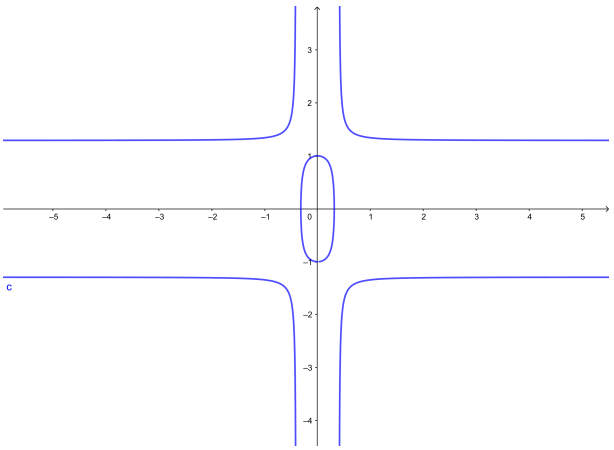

Fig. 6: A twisted Edwards curve of equation $10x^2 + y^2 = 1 + 6x^2y^2$

Montgomery curve

A Montgomery curve over filed K is an elliptic curve defined by an affine equation:

$$by^2 = x(x^2 + ax + 1), \quad (22)$$

where $a^2 \neq 4$ and $b \neq 0$ are parameters in K [25].

Let M be an Montgomery curve over a finite field K. Let $M_1 = (x_1, y_1)$ and $M_2 = (x_2, y_2)$ be points on M. We then define $M_3 = (x_3, y_3) = M_1 + M_2$ [26] where:

$$x_3 = \frac{B(x_2y_1 - x_1y_2)^2}{x_1x_2(x_2 - x_1)^2}, \quad (23)$$

$$y_3 = \frac{(2x_1 + x_2 + A)(y_2 - y_1)}{x_2 - x_1} - \frac{B(y_2 - y_1)^2}{(x_2 - x_1)}. \quad (24)$$

and similarly define $M_4 = (x_4, y_4) = 2M_1$ as:

$$x_4 = \frac{(x_1^2 - 1)^2}{4x_1(x_1^2 + Ax_1 + 1)} \quad (25)$$

$$y_4 = \frac{(2x_1 + x_1 + A)(3x_1^2 + 2Ax_1 + 1)}{2By_1} + \\ - \frac{B(3x_1^2 + 2Ax_1 + 1)^3}{(2By_1)^3} - y_1. \quad (26)$$

Montgomery's curves and twisted Edward's curves cover the same set of elliptic curves. More precisely, for each Montgomery's curve there is a birationally equivalent twisted Edward's curve, and vice versa. Here a birational equivalence between two elliptic curves M , E is a pair of rational maps $M \to E$ and $E \to M$ that are correctly defined almost everywhere [27].

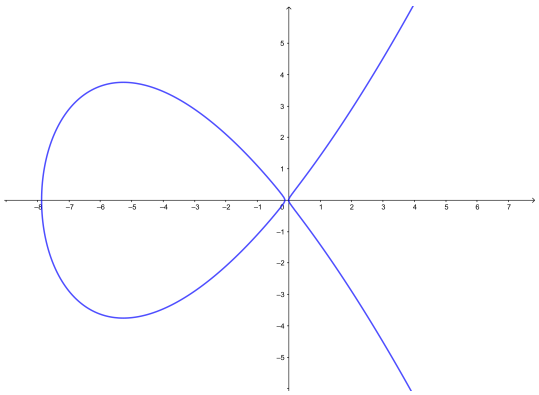

Fig. 7: Montgomery curve of equation $5y^2 = x^3 + 8x^2 + x$.

Hessian curves

A Hessian curve over a field K is given by the cubic equation:

$$x^3 + y^3 + 1 = dxy, \quad (27)$$

for some d with $d^3 \neq 27$ [28].

Let c, d be elements of K such that $c \neq 0$ and $d^3 \neq 27c$. **The generalized Hessian curve** over K is defined by the equation:

$$x^3 + y^3 + c = dxy. \quad (28)$$

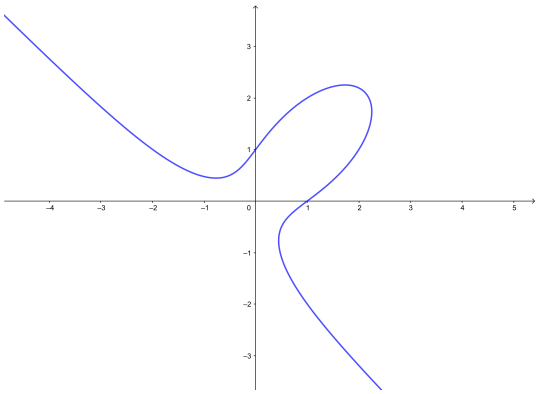

Fig. 8: Hessian curve $x^3 + y^3 = 1 + 4xy$.

The sum of two (different) points $(x_1, y_1), (x_2, y_2)$ is the point (x_3, y_3) given by:

$$x_3 = \frac{y_1^2 x_2 - y_2^2 x_1}{x_2 y_2 - x_1 y_1}; \quad y_3 = \frac{x_1^2 y_2 - x_2^2 y_1}{x_2 y_2 - x_1 y_1}. \tag{29}$$

The doubling of the point (x_1, y_1) is the point (x_3, y_3) given by:

$$x_3 = \frac{y_1(c - x_1^3)}{x_1^3 - y_1^3}; \quad y_3 = \frac{x_1(c - y_1^3)}{x_1^3 - y_1^3}. \tag{30}$$

Theorem 1. *Let E be an elliptic curve over a field K. If the group $E(K)$ has a point of order 3 then E is isomorphic over K to a generalized Hessian curve. Moreover, if K has an element ω with $\omega^2 + \omega + 1 = 0$, then the group $E(K)$ has a point of order 3 if and only if E is isomorphic over K to a generalized Hessian curve.*

The proof of the theorem can be found in [28].

Jacobi quartic curves

A Jacobi quartic form elliptic curve over K is defined by

$$y^2 = x^4 + 2ax^2 + 1 \tag{31}$$

where $a \in K$ with $a^2 \neq 1$. Birational maps between Weierstrass and Jacobi quartic curves can be found in [29].

Let J be a Jacobi quartic curve over a finite field K. Just as before let M_1 and M_2 be points on M. The point M_3 has coordinates:

$$\begin{cases} x_3 = \dfrac{x_1 y_2 + y_1 x_2}{1 - x_1^2 x_2^2} \\ \\ y_3 = \dfrac{(y_1 y_2 + 2a x_1 x_2)(x_1^2 x_2 + 1) + 2x_1 x_2 (x_1^2 + x_2^2)}{(1 - x_1^2 x_2^2)^2} \end{cases}.$$

The identity element is the point $(0, 1)$. The negative of a point (x, y) is $(-x, y)$ [30].

Doubling-oriented Doche – Icart – Kohel curves

Let K be a field and let $a \in K$. Then, the Doubling-oriented Doche – Icart – Kohel curve with parameter a in affine coordinates is represented by:

$$y^2 = x^3 + ax^2 + 16ax \tag{32}$$

This curve is a special case of Weierstrass form. The curve was introduced in 2006 Doche - Icart - Kohel. The parameter a is required to have $a(a-64) \neq 0$. The neutral element of the curve is the unique point at infinity.

Comparison of computing costs in filed $K = \mathbb{F}_p$.

Table 1 compares the computational costs related to the operation of adding or doubling points. A field inversion is abbreviated by I, a multiplication in $\mathbb{F}_p$ by M. It was assumed that the costs of addition and multiplication are the same, and $I = 100M$.

Table 1: Comparison of computing costs in filed $\mathbb{F}_p$ [24, 30].

curve	ADD	DBB
Short Weierstrass	14	11
Edwards	10	7
Twisted Edwards	9	8
Montgomery	-	4
Hessian	12	9
Jacobi quartic	14	7
Doche – Icart – Kohel	17	7

Comparison of ECC and RSA

Tables 2 and 3 compares the elliptic curve cryptography (ECC) algorithm and the Rivest – Shamir – Adleman (RSA) algorithm.

Table 2: Key size ratio for RSA/DSA and ECC with equivalent security level [31].

Key size		Key size
RSA/DSA	**ECC**	**ratio**
1024	16	7:1
2048	224	10:1
3072	256	12:1
7680	384	20:1
15360	521	30:1

The use of ECC in devices with limited resources has a significant advantage over RSA. Cryptography based on elliptical curves gives greater security guarantees and at the same time reduces computational costs, but requires constant improvement to satisfy the limitations of newly designed systems.

Table 3: Security level(bits) and ratio of cost for RSA/DSA and ECC with equivalent security level [31].

Key size		Security Level (bits)	Ratio of Cost
RSA/DSA	ECC		
1024	16	80	3:1
2048	224	112	6:1
3072	256	128	10:1
7680	384	192	32:1
15360	521	256	64:1

Summary

The work discusses groups of curves useful in cryptography. Many of these curves also have their representations in pro- jecting spaces. As shown above, elliptic curve cryptography is a viable alternative to RSA. The continuous development of the popularity of IoT devices requires searching for new solutions in order to increase security while reducing energy consumption.

Literature

[1] Maleszewski W. Algebraic geometry in cryptography at the turn of the xx–xxi century. *Polish Journal of Applied Sciences*, 2(1):11–15, 2017.

[2] Rot A., Blaicke B. Bezpieczeństwo internetu rzeczy. wybrane zagrożenia i sposoby zabezpieczeń na przykładzie systemów produkcyjnych. *Zeszyty Naukowe Politechniki Częstochowskiej. Zarządzanie*, (26):188–198, 2017.

[3] De Mello A. Iiot technologies integration creates growth opportunities in the industrial cybersecurity industry. https://www.iot-now.com/2018/11/28.

[4] Liu X., Yang Y., Choo K.-K.R., Wang H. Security and privacy challenges for internet-of-things and fog computing. *Wireless Communications and Mobile Computing*, 2018.

[5] https://en.wikipedia.org/wiki/Microcontroller.

[6] Stuecheli J., Starke W.J., Irish JD., Arimilli LB¿, Dreps D., Blaner B., Wollbrink C., Allison B. IBM power opens up a new era of acceleration enablement: Opencapi. *IBM Journal of Research and Development*, 62(4/5):8–1, 2018.

[7] https://en.wikipedia.org/wiki/RSA˙(cryptosystem).

[8] Rivest R.L., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[9] Apostol T.M. *Introduction to analytic number theory*. Springer Science & Business Media, 2013.

[10] http://www.crypto-it.net/pl/asymetryczne/rsa.html?tab=1.

[11] Schindler W. A timing attack against rsa with the chinese remainder theorem. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 109–124. Springer, 2000.

[12] Koblitz N. Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209, 1987.

[13] Maleszewski W. Algebraic Geometry in Cryptography at the turn of the XX–XXI Century. *Polish Journal of Applied Sciences*, 2(1):11–15, 2017.

[14] Miller V.S. Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques*, pages 417–426. Springer, 1985.

[15] Liu Z., Großschädl J., Hu Z., Järvinen K., Wang H., Verbauwhede I. Elliptic curve cryptography with efficiently computable endomorphisms and its hardware implementations for the internet of things. *IEEE Transactions on Computers*, 66(5):773–785, 2017.

[16] Sughasiny M. GIVE-AND-TAKE KEY PROCESSING for Cloud-linked IoT.

[17] Johnson D., Menezes A., Vanstone S. The elliptic curve digital signature algorithm (ecdsa). *International journal of information security*, 1(1):36–63, 2001.

[18] Peters Ch., Lange T., Bernstein D.J. Curves, codes, and cryptography. 2011.

[19] https://sefiks.com/2018/12/19/a-gentle-introduction-to-edwards-curves.

[20] Saraf A. *A Study of Edwards Curves in Relation to Elliptic Curve Cryptography*. PhD thesis, Doctoral dissertation, Sri Sathya Sai Institute of Higher Learning, 2015.

[21] Bernstein D., Lange T. Inverted edwards coordinates. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 20–27. Springer, 2007.

[22] Bernstein D., Birkner P., Lange T., Peters Ch. Ecm using edwards curves. *Mathematics of Computation*, 82(282):1139–1179, 2013.

[23] Bernstein D.J., Lange T. Faster addition and doubling on elliptic curves. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 29–50. Springer, 2007.

[24] http://hyperelliptic.org/EFD/g1p/index.html.

[25] Costello C., Smith B. Montgomery curves and their arithmetic. *Journal of Cryptographic Engineering*, 8(3):227–240, 2018.

[26] https://en.wikipedia.org/wiki/Montgomery˙curve.

[27] Bernstein D.J, Lange T. Montgomery curves and the montgomery ladder. *IACR Cryptology ePrint Archive*, 2017:293, 2017.

[28] Farashahi R.R., Joye M. Efficient arithmetic on hessian curves. In *International Workshop on Public Key Cryptography*, pages 243–260. Springer, 2010.

[29] Billet O., Joye M. The jacobi model of an elliptic curve and side-channel analysis. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 34–42. Springer, 2003.

[30] Hisil H., Koon-Ho Wong K., Carter G., Dawson E. Faster group operations on elliptic curves. In *Proceedings of the Seventh Australasian Conference on Information Security-Volume 98*, pages 7–20. Australian Computer Society, Inc., 2009.

[31] Bafandehkar M., Yasin S.M., Mahmod R., Hanapi Z.M. Comparison of ecc and rsa algorithm in resource constrained devices. In *IT Convergence and Security (ICITCS), 2013 International Conference on*, pages 1–3. IEEE, 2013.

Received: 2018
Accepted: 2018