**IB Subject(s): Mathematics**

# Methods for Digital Signature

**Research Question**

**How do digital signatures secure the exchange of information?**

Word Count: $\leq 4000$

# Contents

# 1 Introduction

## 1.1 Cryptography

The starting point of public-key cryptography which currently exists in nearly all of our electronic equipment was generally attributed to a paper written by Diffie-Hellman in 1976. Per this paper, cryptography is the study of mathematical systems for solving two kinds of security problems which are privacy and authentication [1]. Privacy is a method of creating a secure communication between sender and receiver while hiding information from third parties. Authentication, on the other hand, is a method of assuring the receiver of information that the sender is legitimate. Authentication can be thought as a signature that verifies the legitimacy of the document and the person who signed it. The focus of this essay will be on authentication part of cryptography. This essay aims to explore how digital signatures authenticate information by examining the mathematics of two different signature algorithms.

## 1.2 Public-Key Cryptography and Digital Signatures

Public key cryptography has two operations which are encryption and decryption. While encryption process is open to public, decryption process is hidden by the user as a secret[1]. The first successful implementation of public-key cryptography was made by R.L. Rivest, A. Shamir, and L. Adleman in 1978 [2] whose approach is today known as RSA cryptography. ECC(Elliptical Curve Cryptography) was introduced by Victor Miller and Neal Koblitz in 1985 who both independently developed the idea. RSA and ECC are today's most popular public-key cryptography systems.

For E is the encryption process and D is the decryption process, public-key cryptography has four properties[2]:

1. Deciphering the encrypted message M results in M,

$$D(E(M)) = M \tag{1}$$

2. Encryption and decryption processes are easy to compute.

3. Publicizing the encryption processes does not reveal decryption.

4. First decrypting the message M and then encrypting will result in the message M itself,

$$E(D(M))) = M \tag{2}$$

To illustrate the public-key cryptography, let's say Bob is the sender and Alice is the receiver. Bob and Alice both has their own encryption and decryption processes which are $E_B, D_B, E_A, D_A$ respectively.

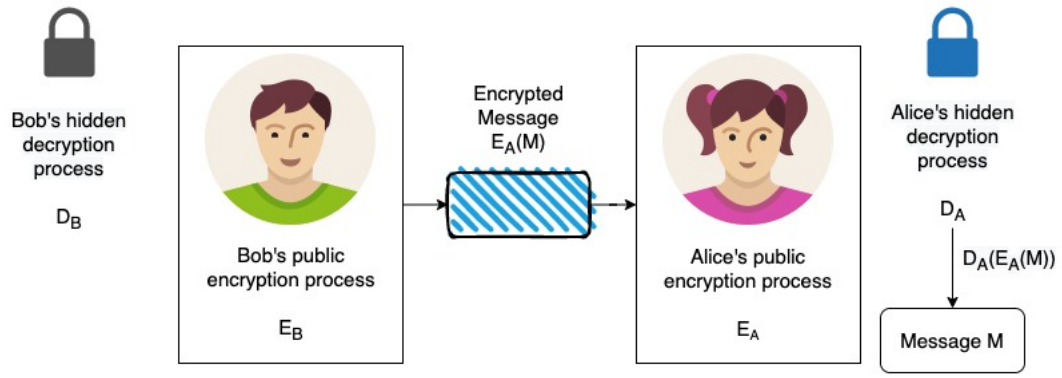



Figure 1: Illustration of Public-Key Cryptography.

Bob uses Alice's public encryption process to secure the message and sends the encrypted message to Alice. Alice can decrypt the message by using her hidden

description process. Since only Alice has the access to her decryption method, no third unauthorized party can access the message. So thus privacy is protected.

However, in the case of authentication, Bob decrypts the message by his hidden decryption process $D_B$ and sends the message as $D_B(M)$, which is also Bob's signature $S = D_B(M)$. When Alice receives the signature, by the rule of the fourth property of public-key cryptography $E(D(M))) = M$, Alice uses Bob's encryption process to learn the message $E_B(D_B(M))) = M$. So now, Alice has received two texts from Bob: the signature $S$ and the message $M$. The pair of $(M, S)$ constitutes a signed message from Bob, since only Bob can use his decryption process to produce the signature $S$.

In this paper, first RSA Digital Signature Algorithm then Elliptical Curve Digital Signature Algorithm will be explored through proofs and their applications. In the end, both algorithms will be compared to decide on which one is more suitable for the growing technologies such as internet of things devices and cryptocurrencies.

# 2 RSA Digital Signature Algorithm

## 2.1 RSA

RSA cryptography is based on modular arithmetic and prime numbers. The strength of RSA comes from the difficulty in factoring large numbers into their prime factors.

## 2.2 The Underlying Mathematics

There are three fundamental definitions that should be considered before diving into mathematics of RSA: prime numbers, greatest common divisor and relatively prime numbers.

**Definition 1.** A prime number, $p \in \mathbb{Z}^+$, is a number which has only 2 positive factors which are 1 and itself(the number 1 is not a prime so it is excluded from the definition).

**Definition 2.** Greatest common divisor is the greatest positive integer, $Z^+$, that can divide both of integers a and b in the condition that $a, b \neq 0$,expressed as $gcd(a, b)$.

**Definition 3.** Relatively prime numbers are positive integers which do not have a common factor other than 1 that is for integers $a$ and $b$ $gcd(a, b) = 1$.

### 2.2.1 Euler Totient Function

*Euler Totient Function* $\varphi(n)$ counts the number of integers in between 1 and $n$ that are coprime with $n$.[4] In other words, for positive integers $k$ and $n$, $\varphi(n)$

is a function that returns the number of $k$ which is $k < n$ and $gcd(n, k) = 1$.

**Example 1.** In order to find $\varphi(9) = 6$, positive integers that are both less than 9 and relatively prime with 9 should be counted.

$$gcd(1,9) = 1 \qquad gcd(4,9) = 1 \qquad gcd(7,9) = 1$$
$$gcd(2,9) = 1 \qquad gcd(5,9) = 1 \qquad gcd(8,9) = 1$$
$$gcd(3,9) = 3 \qquad gcd(6,9) = 3$$

As can be seen only 6 of the positive integers less than 9 are relatively prime with 9, resulting in $\varphi(9) = 6$. Let's look at another example of Euler totient function.

**Example 2.** In order to find $\varphi(7) = 6$, positive integers that are both less than 7 and relatively prime with 7 should be counted.

$$gcd(1,7) = 1 \qquad gcd(4,7) = 1$$
$$gcd(2,7) = 1 \qquad gcd(5,7) = 1$$
$$gcd(3,7) = 1 \qquad gcd(6,7) = 1$$

| $n$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\varphi(n)$ | 1 | 1 | 2 | 2 | 4 | 2 | 6 | 4 | 6 | 4 | 10 | 4 |

Table 1: Outputs of Euler Totient Function for first 12 positive integers

For a prime number $p$, since all the positive integers less than $p$, $1 \leq n < p$, are coprime with $p$ due to the definiton, we can hypothesize that Euler Totient function for a prime number $p$ equals $p-1$. For example in Example 2, 7, a prime number, returned one less of its value through Euler Totient Function,6. When Table 1 is reviewed, it seems to be supporting our hypothesis that all the prime numbers returns one less of themselves in the Euler Totient Function.

Therefore, we conclude that for a prime number $p$,

$$\varphi(p) = p - 1 \tag{3}$$

Desmos is used to plot first the 250 values of Euler Totient Function.

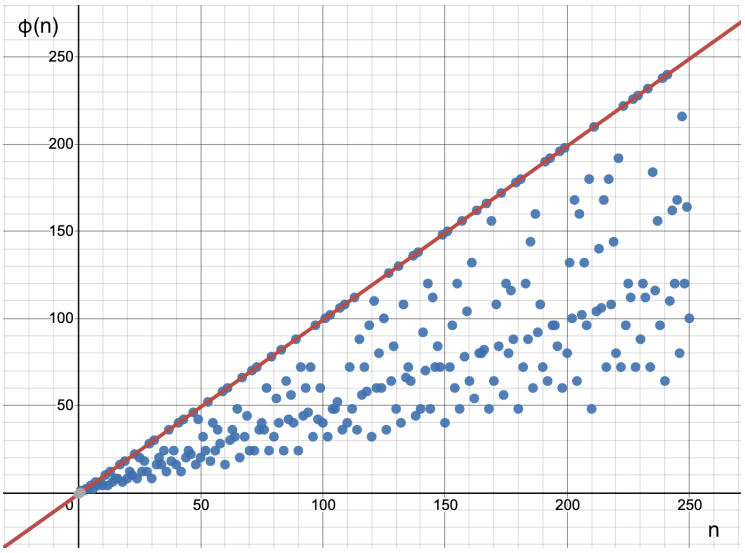



Figure 2: Graph of Euler Totient Function with x axis $n$, y axis $\varphi(n)$.

The upper boundary for the graph is $y = x - 1$ shown with red line is due to the fact that the highest values Euler Totient Function can return is at prime numbers which result in $\varphi(p) = p - 1$. So, we can conclude that the all the numbers that lie on the red line are prime numbers. In RSA cryptography, these prime numbers will be mainly in consideration.

It is important to consider that Totient function of a positive integer n is equal to the multiplication of the Totient function values of its prime factors. Formally, for two different prime numbers $a$ and $b$,

$$\varphi(a \cdot b) = \varphi(a) \cdot \varphi(b) = (a - 1)(b - 1) \tag{4}$$

**Proof.** Positive integer $n = a \cdot b$ has two prime factors $a$ and $b$. A number is coprime with every number which are not multiples of its prime factors. Therefore, to find $\varphi(n)$ number of coprime positive integers of $n$ which are smaller than $n$, we can subtract the multiples of $a$ and $b$ from $n$. There are $a \cdot b$ numbers including 1 and $n$.

$a \cdot b$ is not coprime with the multiples of $a$ and $b$. The number of multiples of $a$ smaller than or equal to $n$ is $b$, while the number of multiples of $b$ smaller than or equal to $n$ is $a$. However, when the number of multiples are subtracted from $n$, since $n = a \cdot b$ is counted as multiple twice(for both $a$ and $b$), 1 should be added to find the Totient function. So the number of coprimes with $n$ smaller than $n(\varphi(n = a \cdot b))$ is,

$$\varphi(a \cdot b) = a \cdot b - a - b + 1 \tag{5}$$

$$= (a - 1)(b - 1) \tag{6}$$

$$= \varphi(a) \cdot \varphi(b) \tag{7}$$

**Example.** In order to find $\varphi(10)$, find its prime factors: 2,5 and then multiply the totient function of their primes $\varphi(10) = (2 - 1)(5 - 1) = 1 \cdot 4 = 4$. Table 1 confirms this result.

### 2.2.2 Modular Arithmetic

Modular arithmetic, in the most basic definition, is an operation that gives the remainder when a number is divided by a particular divisor. If the remainders of two integers $a$ and $b$ when divided by another integer $n$ are equal to each other,

they are called congruent/equivalent and told to be equivalent modulo $n$.

$$\frac{a-b}{n} = k \in \mathbb{Z}^+ \tag{8}$$

$$a - b = kn, n \in \mathbb{Z}, n > 0 \tag{9}$$

$$a \equiv b(mod\, n) \Leftrightarrow n|a - b \tag{10}$$

For example, when 38 and 14 are divided by 12, they give the same remainder. Therefore, 38 and 14 are equivalent in modulo 12.

$$38 \equiv 2(mod\, 12) \tag{11}$$

$$14 \equiv 2(mod\, 12) \tag{12}$$

Modular arithmetic has 3 basic properties. For $a, b, c, d \in \mathbb{Z}$ and $n > 1$,

1. If $a \equiv b(mod\, n)$ and $b \equiv c(mod\, n)$, then $a \equiv c(mod\, n)$
2. If $a \equiv b(mod\, n)$ and $c \equiv d(mod\, n)$, then $a + c \equiv b + d(mod\, n)$
3. If $a \equiv b(mod\, n)$ and $c \equiv d(mod\, n)$, then $a \cdot c \equiv b \cdot d(mod\, n)$

In modular arithmetic, there are modular inverses.

**Definition 4.** Given an integer $a$ with $gcd(a, m) = 1$, an integer solution of $x$ of $ax \equiv 1(mod\, m)$ is called an inverse of $a$ modulo m.[3] That is, when $a$ and $m$ are relatively prime, a modular multiplicative inverse is unique. It is generally denoted as, $x \equiv a^{-1} \equiv \frac{1}{a}(mod\, m)$.

### 2.2.3 Chinese Remainder Theorem

The proof of RSA in the next section uses Chinese Remainder Theorem. Therefore, now, this theorem will be given and proven.

**Theorem 1: Chinese Remainder Theorem** Let $p_1$ and $p_2$ be relatively prime positive integers. Then the system of congruences

$$x \equiv a_1 (mod\, p_1) \tag{13}$$

$$x \equiv a_2 (mod\, p_2) \tag{14}$$

has a unique solution in modulo $M = p_1 p_2$. If $a_1 = a_2 = k$, then solution for this system of congruences is $x \equiv k (mod\, p_1 p_2)$.

In the proof of Chinese Remainder Theorem, **Bezout's Identity** will be used. Bezout's identity states that for integers $a$ and $b$ [3],

$$ma + nb = gcd(a, b) \tag{15}$$

**Proof.** Firstly, proof by contradiction will be used to prove the uniqueness of the solution.

Assume that $x_1$ and $x_2$ are two different solutions to a system of congruences. Since they are both solutions for the system of congruences, $x_1 - x_2 \equiv 0 (mod\, p_1)$ and $x_1 - x_2 \equiv 0 (mod\, p_2)$. Therefore, $p_1 | (x_1 - x_2)$ and $p_2 | (x_1 - x_2)$. As the theorem states $gdc(p_1, p_2) = 1$, then $p_1 p_2 | x_1 - x_2$. As a result,

$$x_1 - x_2 \equiv 0 (mod\, p_1 p_2) \tag{16}$$

$$x_1 \equiv x_2 (mod\, p_1 p_2) \tag{17}$$

This equivalence contradicts the first claim that there are two different solutions, proving the uniqueness of the solution.

Now, Bezout's identity will be used to find the solution. The greatest common divisor of two relatively prime numbers $p_1$ and $p_2$ is 1 due to the definition. For $gcd(p_1, p_2) = 1$, Bezout's identity implies that $m_1p_1 + m_2p_2 = gcd(p_1, p_2) = 1$. Finding the $m_1p_1 + m_2p_2$ in modulo $p_1$ and in modulo $p_2$ respectively, we reach the following,

$$m_1p_1 + m_2p_2 \equiv m_2p_2 \equiv 1(mod\, p_1) \tag{18}$$

$$m_1p_1 + m_2p_2 \equiv m_1p_1 \equiv 1(mod\, p_2) \tag{19}$$

Remembering the definition of modular inverse, since $gdc(p_1, p_2) = 1$, $m_1$ is modular inverse of $p_1$ modulo $p_2$ and $m_2$ is modular inverse of $p_2$ modulo $p_1$. If both sides of the equivalence is multiplied with the coefficients $a_1$ and $a_2$, we reach the solution,

$$x \equiv a_1 \cdot m_2p_2 \equiv a_1(mod\, p_1) \tag{20}$$

$$x \equiv a_1 \cdot m_1p_1 \equiv a_2(mod\, p_2) \tag{21}$$

So the solution is,

$$x = a_1m_2p_2 + a_2m_1p_1 \tag{22}$$

$$x \equiv a_1m_2p_2 + a_2m_1p_1(mod\, p_1p_2) \tag{23}$$

If $a_1 = a_2 = k$, then,

$$x = km_2p_2 + km_1p_1 = k(m_1p_1 + m_2p_2) \tag{24}$$

Due to Bezout's identity, we already know that $m_1p_1 + m_2p_2 = 1$, yielding the solution as,

$$x \equiv k(mod\, p_1p_2) \tag{25}$$

### 2.2.4 Fermat's Little Theorem

**Theorem 3. Fermat's Little Theorem.** If $p$ is prime and $a$ is an integer while $a$ cannot be divided by p, then $a^{p-1} \equiv 1(mod\, p)$[3]. By the third property of modular arithmetic, both sides of the equivalence can be multiplied by $a$ to rewrite the equivalence as $a^p \equiv a(mod\, p)$.

**Proof 3.** In order to prove Fermat's Little Theorem, mathematical induction will be used.

For $a = 1$, $1^p \equiv 1(modp)$ where $p$ is a prime number is true since the remainder of 1 when divided by any prime number would be 1 again.

For $a = k$ and $k \in \mathbb{Z}^+$, assume $k^p \equiv k(mod\, p)$ to be true.

Let $a = k + 1$, in order to prove the equivalence of $(k + 1)^p \equiv k + 1(mod\, p)$, binomial expansion will be used,

$$(k+1)^p = k^p + \binom{p}{1}k^{p-1} + \binom{p}{2}k^{p-2} + \ldots + \binom{p}{l}k^{p-w} + \binom{p}{p-1}k + 1 \tag{26}$$

Since every coefficient in the form of $\binom{p}{w}$, $1 \leq w \leq p-1$ can be rewritten as $\binom{p}{w} = \frac{p!}{w!(p-w)!}$, they are divisible by $p$. Therefore, the remainder when $(k + 1)^p$ divided by $p$ is $k^p + 1$. So it can be written as $(k + 1)^p \equiv k^p + 1(mod\, p)$.

Now we have proven the inductive step, the theorem is also proven.

## 2.3 Fundamentals and Proof of RSA

RSA cryptography follows certain steps. In this section, RSA cryptography is split into four main steps: calculating encryption and decryption keys, finding a numeric value for the message, encryption, and decryption.

**1) Calculating Encryption and Decryption Keys:**

First determine two distinct large prime numbers, $p$ and $q$. Then multiply them to reach to value $n = p \cdot q$. Now calculate the Euler Totient Function of $n$,

$$\varphi(n) = (p-1)(q-1) \tag{27}$$

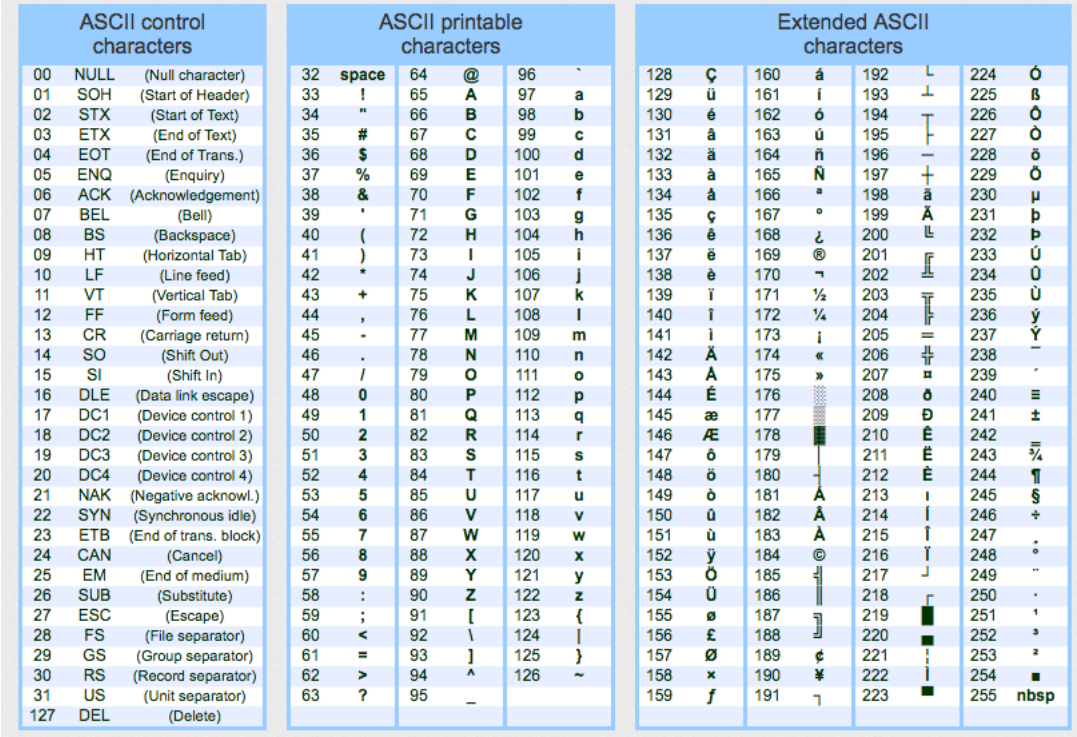
Now that we have calculated the Euler Totient function of n, remove the prime numbers $p$ and $q$ from the system. Determine the encryption key $e$ and the decryption key $d$ so that $e$ and $\varphi(n)$ are coprime and $e$ and $d$ are each other's multiplicative inverse modulo $\varphi(n)$,

$$e \cdot d \equiv 1 (mod\, \varphi(n)) \tag{28}$$

This equivalence concludes that $(e, n)$ is the public encryption key, whereas $d$ is the private decryption key.

**2) Finding a Numeric Value for the Message:**

Since RSA works through modular arithmetic, it is important that the message that will be secured or authenticated should be converted into a numeric form. To that end, ASCII conversion table is used.



ASCII control characters

| | | |
|---|---|---|
| 00 | NULL | (Null character) |
| 01 | SOH | (Start of Header) |
| 02 | STX | (Start of Text) |
| 03 | ETX | (End of Text) |
| 04 | EOT | (End of Trans.) |
| 05 | ENQ | (Enquiry) |
| 06 | ACK | (Acknowledgement) |
| 07 | BEL | (Bell) |
| 08 | BS | (Backspace) |
| 09 | HT | (Horizontal Tab) |
| 10 | LF | (Line feed) |
| 11 | VT | (Vertical Tab) |
| 12 | FF | (Form feed) |
| 13 | CR | (Carriage return) |
| 14 | SO | (Shift Out) |
| 15 | SI | (Shift In) |
| 16 | DLE | (Data link escape) |
| 17 | DC1 | (Device control 1) |
| 18 | DC2 | (Device control 2) |
| 19 | DC3 | (Device control 3) |
| 20 | DC4 | (Device control 4) |
| 21 | NAK | (Negative acknowl.) |
| 22 | SYN | (Synchronous idle) |
| 23 | ETB | (End of trans. block) |
| 24 | CAN | (Cancel) |
| 25 | EM | (End of medium) |
| 26 | SUB | (Substitute) |
| 27 | ESC | (Escape) |
| 28 | FS | (File separator) |
| 29 | GS | (Group separator) |
| 30 | RS | (Record separator) |
| 31 | US | (Unit separator) |
| 127 | DEL | (Delete) |

ASCII printable characters

| | | | | | |
|---|---|---|---|---|---|
| 32 | space | 64 | @ | 96 | ` |
| 33 | ! | 65 | A | 97 | a |
| 34 | " | 66 | B | 98 | b |
| 35 | # | 67 | C | 99 | c |
| 36 | $ | 68 | D | 100 | d |
| 37 | % | 69 | E | 101 | e |
| 38 | & | 70 | F | 102 | f |
| 39 | ' | 71 | G | 103 | g |
| 40 | ( | 72 | H | 104 | h |
| 41 | ) | 73 | I | 105 | i |
| 42 | * | 74 | J | 106 | j |
| 43 | + | 75 | K | 107 | k |
| 44 | , | 76 | L | 108 | l |
| 45 | - | 77 | M | 109 | m |
| 46 | . | 78 | N | 110 | n |
| 47 | / | 79 | O | 111 | o |
| 48 | 0 | 80 | P | 112 | p |
| 49 | 1 | 81 | Q | 113 | q |
| 50 | 2 | 82 | R | 114 | r |
| 51 | 3 | 83 | S | 115 | s |
| 52 | 4 | 84 | T | 116 | t |
| 53 | 5 | 85 | U | 117 | u |
| 54 | 6 | 86 | V | 118 | v |
| 55 | 7 | 87 | W | 119 | w |
| 56 | 8 | 88 | X | 120 | x |
| 57 | 9 | 89 | Y | 121 | y |
| 58 | : | 90 | Z | 122 | z |
| 59 | ; | 91 | [ | 123 | { |
| 60 | < | 92 | \ | 124 | \| |
| 61 | = | 93 | ] | 125 | } |
| 62 | > | 94 | ^ | 126 | ~ |
| 63 | ? | 95 | _ | | |

Extended ASCII characters

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 128 | Ç | 160 | á | 192 | └ | 224 | Ó |
| 129 | ü | 161 | í | 193 | ┴ | 225 | ß |
| 130 | é | 162 | ó | 194 | ┬ | 226 | Ô |
| 131 | â | 163 | ú | 195 | ├ | 227 | Ò |
| 132 | ä | 164 | ñ | 196 | ─ | 228 | õ |
| 133 | à | 165 | Ñ | 197 | ┼ | 229 | Õ |
| 134 | å | 166 | ª | 198 | ã | 230 | µ |
| 135 | ç | 167 | º | 199 | Ã | 231 | þ |
| 136 | ê | 168 | ¿ | 200 | ╚ | 232 | Þ |
| 137 | ë | 169 | ® | 201 | ╔ | 233 | Ú |
| 138 | è | 170 | ¬ | 202 | ╩ | 234 | Û |
| 139 | ï | 171 | ½ | 203 | ╦ | 235 | Ù |
| 140 | î | 172 | ¼ | 204 | ╠ | 236 | ý |
| 141 | ì | 173 | ¡ | 205 | ═ | 237 | Ý |
| 142 | Ä | 174 | « | 206 | ╬ | 238 | ¯ |
| 143 | Å | 175 | » | 207 | ¤ | 239 | ´ |
| 144 | É | 176 | ░ | 208 | ð | 240 | ≡ |
| 145 | æ | 177 | ▒ | 209 | Ð | 241 | ± |
| 146 | Æ | 178 | ▓ | 210 | Ê | 242 | ‗ |
| 147 | ô | 179 | │ | 211 | Ë | 243 | ¾ |
| 148 | ö | 180 | ┤ | 212 | È | 244 | ¶ |
| 149 | ò | 181 | Á | 213 | ı | 245 | § |
| 150 | û | 182 | Â | 214 | Í | 246 | ÷ |
| 151 | ù | 183 | À | 215 | Î | 247 | ¸ |
| 152 | ÿ | 184 | © | 216 | Ï | 248 | ° |
| 153 | Ö | 185 | ╣ | 217 | ┘ | 249 | ¨ |
| 154 | Ü | 186 | ║ | 218 | ┌ | 250 | · |
| 155 | ø | 187 | ╗ | 219 | █ | 251 | ¹ |
| 156 | £ | 188 | ╝ | 220 | ▄ | 252 | ³ |
| 157 | Ø | 189 | ¢ | 221 | ¦ | 253 | ² |
| 158 | × | 190 | ¥ | 222 | Ì | 254 | ■ |
| 159 | ƒ | 191 | ┐ | 223 | ▀ | 255 | nbsp |

Figure 3: ASCII Table.

**3) Encryption:**

Using the public key $(e, n)$ and message $m$, for the encrypted message $c$ is a positive integer, encryption process can be completed by the following equation,

$$m^e \equiv c(mod\, n) \tag{29}$$

**4) Decryption:**

Similarly, decryption can be processed by taking the power of decryption key $d$ of the expression,

$$(m^e)^d \equiv c^d \equiv m(mod\, n) \tag{30}$$

**Proof of RSA**

RSA cryptography holds that for natural numbers $m, e, d, n \in \mathbb{Z}^+$ and prime numbers $p$ and $q$,

$$n = pq \tag{31}$$

$$e \cdot d \equiv 1 (mod\, \varphi(n)) \tag{32}$$

$$(m^e)^d \equiv (m^d)^e \equiv m (mod\, n) \tag{33}$$

In order to prove the correctness of RSA, let's use Fermat's Little Theorem defined in the Underlying Mathematics section. Fermat's Little Theorem asserts that,

$$a^p \equiv a (mod\, p) \tag{34}$$

By multiplying each side of the equivalence by $a$, equation 19 can be written as,

$$a^{p-1} \equiv 1 (mod\, p) \tag{35}$$

Now, by the rule of multiplication of modular arithmetic, let's multiply equation 19 and equation 20,

$$a^{p-1} \cdot a^p \equiv a (mod\, p) \tag{36}$$

Repeat this multiplication with $a^{p-1}$ for $D$ times,

$$a^{D(p-1)} \cdot a^p \equiv a (mod\, p) \tag{37}$$

Let's gather the powers on a common base,

$$a^{D(p-1)} \cdot a^{p-1} \cdot a^1 \equiv a(mod\, p) \tag{38}$$

$$a^{(D+1)(p-1)} \cdot a^1 \equiv a(mod\, p) \tag{39}$$

$$a^{(D+1)(p-1)+1} \equiv a(mod\, p) \tag{40}$$

In order to prove the correctness of RSA, let's use the symbol for message $m$ for $a$ as stated in the thesis of original paper [2].

$$m^{(D+1)(p-1)+1} \equiv m(mod\, p) \tag{41}$$

According to equation 16 $n = pq$. Since p and q are prime numbers, their Euler Totient function equals to $p-1$ and $q-1$,respectively. Therefore, the Euler function of their multiplication will be equal to the multiplication of their individual totient function values. That is,

$$\varphi(n) = (p-1)(q-1) \tag{42}$$

According to equation 17 $ed \equiv 1(mod\, \varphi(n))$. Let's combine these equations,

$$e \cdot d \equiv 1(mod\, (p-1)(q-1)) \tag{43}$$

This congruence can be written as an equation, for $L$ is a positive integer,

$$e \cdot d = k(p-1)(q-1) + 1 \tag{44}$$

Returning back to equation 27, we can substitute $k(p-1) = D+1$ into equation

27. At this point, the power becomes equal to the equation 30,

$$m^{k(p-1)(p-1)+1} \equiv m(mod\, p) \tag{45}$$

$$m^{ed} \equiv m(mod\, p) \tag{46}$$

We can find the same equivalence in equation 32 by repeating the process for $q$ instead of $p$, which will result in,

$$m^{ed} \equiv m(mod\, q) \tag{47}$$

Now that we have two equivalences, in order to combine them with a same modulo, we can use Chinese Remainder Theorem explored previously. According to Chinese Remainder Theorem,

$$m^{ed} \equiv m(mod\, p) \ \wedge \ m^{ed} \equiv m(mod\, q) \implies m^{ed} \equiv m(mod\, pq) \tag{48}$$

Substituting $n = pq$ into equation 33, the correctness of RSA is proven,

$$m^{ed} \equiv m(mod\, n) \tag{49}$$

## 2.4 Application of RSA Digital Signature

To illustrate a real-life application of RSA Digital Signature, let's consider a case. IBO wants to share the starting day of May 22 exams with all the IB school coordinators all around the world. Since this process will be very hard with paper-signed documents, IBO decides to use RSA digital signature to officially share the starting day of exams which is April 28.

The first thing IBO does is to determine to prime numbers, for which they decide to use $p = 13$ and $q = 7$. Normally, very large prime numbers should be chosen, but for the sake of illustration these numbers are chosen. Then, they calculate the $n$ value through $n = p \cdot q = 13 \cdot 7 = 91$ and Euler Totient function through $\varphi(91) = (13 - 1)(7 - 1) = 72$.

In order to calculate their encryption and decryption keys, they need to find a number that has 2 prime factors and which is equivalent to 1 modulo 72 that is $ed \equiv 1(mod\, 72)$. They consider the numbers that are 1 modulo 72: 73, 145, 217, 289 ... Since a number with 2 prime factors should be chosen they decide on $217 = 7 \cdot 31$. Either of the prime factors can be encryption or decryption key. They decide that their encryption key $(e, n)$ is $(31, 91)$ and decryption key $d$ is 7.

Now, they have to convert 'APRIL28' into the numeric form by ASCII using Figure 3 and then decrypt the message by taking the dth(7th) power of the message modulo n(91).

| character | 2 | 8 | A | P | R | I | L |
|---|---|---|---|---|---|---|---|
| ASCII form | 50 | 56 | 65 | 80 | 82 | 73 | 76 |

Table 2: ASCII Conversion of 'APRIL28'

To decrypt the message in ASCII form, 7th power of each character modulo 91 is taken. For Calculations, TI-Inspire is used,

$$
\begin{aligned}
2 &\implies 50^7 \equiv 15 (mod\, 91) \\
8 &\implies 56^7 \equiv 56 (mod\, 91) \\
A &\implies 65^7 \equiv 65 (mod\, 91) \\
P &\implies 80^7 \equiv 24 (mod\, 91) \\
R &\implies 82^7 \equiv 82 (mod\, 91) \\
I &\implies 73^7 \equiv 31 (mod\, 91) \\
L &\implies 76^7 \equiv 41 (mod\, 91)
\end{aligned}
$$

So the signature is $S = 15\ 56\ 65\ 24\ 82\ 31\ 41$. IBO shares this signature and their public key $(e, n)$ with all IB coordinators. Any IB coordinator now can find the message from this signature by taking encryption key eth(31th) power of the integers in the signature modulo n(91),

$$
\begin{aligned}
15^{31} (mod\, 91) \equiv 50 &\implies 2 \\
56^{31} (mod\, 91) \equiv 56 &\implies 8 \\
65^{31} (mod\, 91) \equiv 65 &\implies A \\
24^{31} (mod\, 91) \equiv 80 &\implies P \\
82^{31} (mod\, 91) \equiv 82 &\implies R \\
31^{31} (mod\, 91) \equiv 73 &\implies I \\
41^{31} (mod\, 91) \equiv 76 &\implies L
\end{aligned}
$$

Now, IB coordinators have a pair of information: the message and the signature. $(M, S) = ('28APRIL', 15\ 56\ 65\ 24\ 82\ 31\ 41)$ Thus, the date of examination is digitally signed by IBO.

# 3 Elliptical Curve Digital Signature Algorithm

## 3.1 Elliptic Curves

Elliptical Curve Digital Signature Algorithm(ECDSA), as the name suggests, directly depends on the arithmetic of elliptic curves. In cryptography, normally, elliptic curves that are defined in finite fields(which will be explained later on) are used. However, in the beginning, for the sake of simplicity, elliptic curves over real numbers will be explored. Then, a complete definition will be given.

A basic definition of elliptic curves is that elliptic curves are set of points that satisfy the following equation,

$$y^2 = x^3 + ax + b \tag{50}$$

Desmos is used to draw $y^2 = x^3 - 2x + 3$ curve to demonstrate the general shape of elliptic curves that are non-singular.

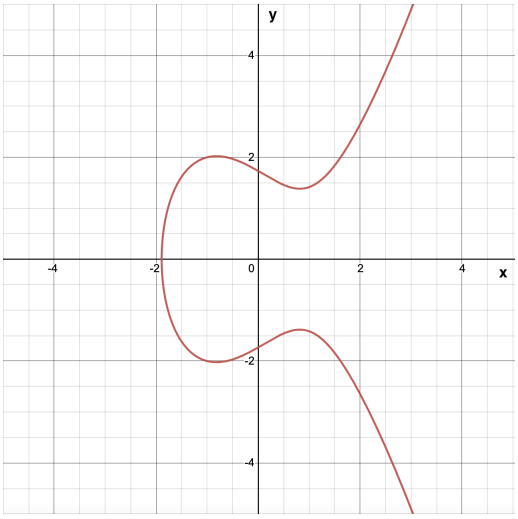



Figure 4: Graph of $y^2 = x^3 - 2x + 3$.

**Definition 5.** A curve $f(x, y)$ is singular in a point $P = (x_p, y_p)$ if and only if $\frac{df}{dx} = \frac{df}{dy} = 0$[9]. If any such point exists, the curve is called as a singular curve.

However, the case of singularity should be prevented because it makes ECC cryptography insecure[9]. Therefore, as a rule, the coefficients $a$ and $b$ are chosen so that $4a^3 + 27b^2 \neq 0$. Let's prove this condition for non-singularity.

**Proof.** Let's define a curve $f(x, y)$ as $f : \ y^2 = x^3 + ax + b$ and take the derivative both with respect to x and y and equal them to 0 to find a singular point having the conditions stated in the definition,

$$\frac{df}{dx} = 3x^2 + a = 0 \tag{51}$$

$$\frac{df}{dy} = 2y = 0 \tag{52}$$

From these two equations, we can conclude that in the condition $x = ((\frac{-a}{3})^{\frac{1}{2}})$ and $y = 0$, there is a singular point. Now, these two points should be substituted into the original curve,

$$0 = (\frac{-a}{3})^{\frac{3}{2}} + a(\frac{-a}{3})^{\frac{1}{2}} + b \tag{53}$$

$$-b = (\frac{-a}{3})^{\frac{3}{2}} + a(\frac{-a}{3})^{\frac{1}{2}} \tag{54}$$

$$b^2 = (\frac{-a}{3})^3 - \frac{a^3}{3} + \frac{2a^3}{9} \tag{55}$$

$$b^2 = \frac{-4a^3}{27} \implies 4a^3 + 27b^2 = 0 \tag{56}$$

So the condition for non-singularity $4a^3 + 27b^2 \neq 0$ is proven.

To illustrate, a singular curve $y^2 = x^3 - 3x + 2$ for which $4 \cdot (-3)^3 + 27 \cdot (2)^2 = 0$ is drawn.

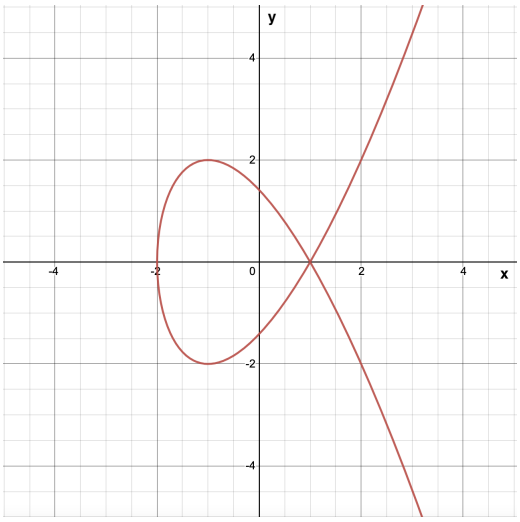



Figure 5: Graph of $y^2 = x^3 - 2x + 3$.

## 3.2 Group Operations

**Definition 6.** A group is a set of elements $\mathbb{G}$ closed under a binary operation $+$ that satisfies the following three axioms[7],

1. The operation $+$ is associative. That is, if $a$ and $b$ are members of the group $\mathbb{G}$, then (a+b)+c = a+(b+c).

2. There is an element $e$ in $\mathbb{G}$ that acts as the identity element for the group. The identity element provides that for $z$ is any element in the group, $e+z = z$ and $z + e = z$.

3. For any element $z$ in the group $\mathbb{G}$, there exists the inverse of the element $z^{-1}$ which provides that $z^{-1} + z = e$, where $e$ is the identity element.

Considering this definition, it can be concluded that set of integer numbers $\mathbb{Z}$ is a group, whereas natural numbers $\mathbb{N}$ is not a group since it does not satisfy the third property.

Now, a group over elliptic curves whose elements are the points of an elliptic

curve can be defined,

1. The inverse of a point $P$ is the point $-P$, one symmetric about the $x-axis$.
2. Identity element is the point at infinity $\infty$ that $P+\infty=\infty+P$. It is also defined as $P+(-P)=\infty$.
3. Addition is calculated by three non-zero points on a same line $P,Q,R$ by the equation $P+Q+R=\infty$.

### 3.2.1 Geometric Addition and Algebraic Addition

According to the addition property $P+Q+R=\infty$. This implies that $P+Q=-R$. Therefore, in order to add two points on the curve geometrically, a line drawn passing through the points to intersect at a third point. The symmetry of the third point in $x$-axis is taken, as can be seen in Figure 6.

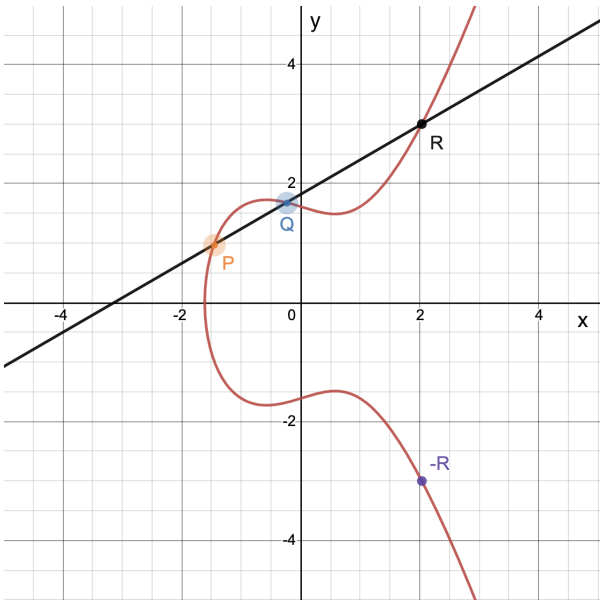



Figure 6: Geometric point addition on an elliptic curve.

In order to add two points algebraically, let $P$ have coordinates $(x_P, y_P)$ and

$Q$ $(x_Q, y_Q)$. For $P \neq Q$ and $x_P \neq y_P$, the line $PQ(y = mx + c)$ has a slope,

$$m_{PQ} = \frac{y_P - y_Q}{x_P - x_Q} \tag{57}$$

The equation of the line can be arranged,

$$y = mx_P + c = mx + (y_P - mx_P) = m(x - x_P) + y_P \tag{58}$$

$$y^2 = (m(x - x_P) + y_P)^2 \tag{59}$$

Set it equal to,

$$(m(x - x_P) + y_P)^2 = x^3 + Ax + B \tag{60}$$

$$x^3 + Ax + B - (m(x - x_P) + y_P)^2 = 0 \tag{61}$$

Expand,

$$x^3 - m^2x^2 + \ldots = 0 \tag{62}$$

Cubic polynomials also can be written as,

$$x^3 + cx^2 + dx + x = (x - x_P)(x - x_Q)(x - x_R) = 0 \tag{63}$$

$$= x^3 - (x_P + x_Q + x_R)x^2 + \ldots = 0 \tag{64}$$

Then,

$$x^3 - (x_P + x_Q + x_R)x^2 + \ldots = x^3 - m^2x^2 + \ldots \tag{65}$$

$$m^2 = (x_P + x_Q + x_R) \tag{66}$$

Thus, $P + Q = -R = (x_R, -y_R)$ is calculated,

$$m^2 = (x_P + x_Q + x_R) \implies x_R = m^2 - x_P - x_Q \tag{67}$$

$$y^R = m(x_R - x_P) + y_P \implies -y_R = (x_P - x_R) - y_P \tag{68}$$

A special case for $P + Q = -R$ is $x_P = x_R$ and $y_P = -y_Q$ which implies that $P + (-P) = \infty$ as shown in Figure 7.

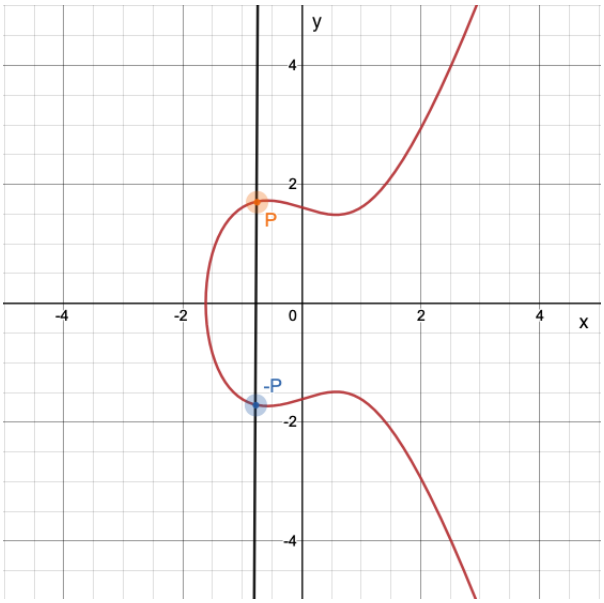



Figure 7: Identity Element $P + (-P) = \infty$.

### 3.2.2 Geometric Doubling and Algebraic Doubling

Point doubling is very similar to addition with a basic difference that two same points are added $P + P = 2P$.

Since same points are added, the line drawn is tangent to the curve at point $P$. Geometrically, the operation is shown in Figure 8.

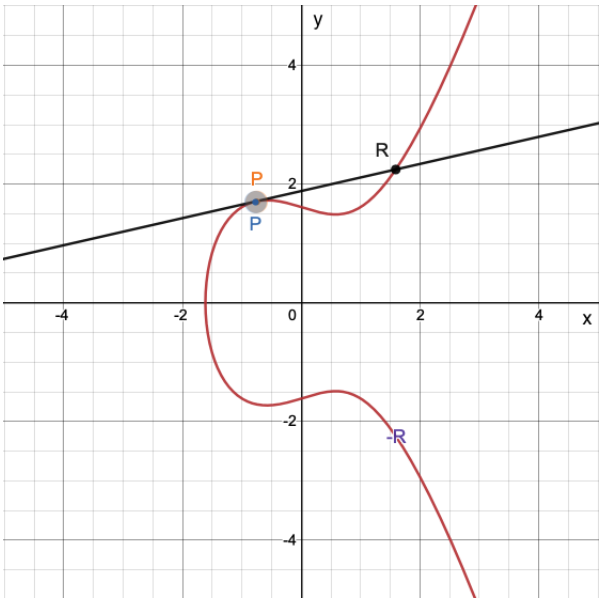



Figure 8: Geometric point doubling $2 \times P = 2G$.

As seen in Figure 8, the line $PR$ is tangent to the curve at point $P = (x_P, y_P)$. The derivative of the elliptic curve function with respect to $x$ is the gradient function of the tangent line,

$$y^2 = x^3 + Ax + B \tag{69}$$

$$\frac{d}{dx}(y^2) = \frac{d}{dx}(x^3 + Ax + B) \tag{70}$$

$$2y\frac{dy}{dx} = 3x^2 + A \tag{71}$$

$$m = \frac{dy}{dx} = \frac{3x^2 + A}{2y} \tag{72}$$

The slope of the tangent line at point $P$ can be found by substituting the coordinates of $P = (x_P, y_P)$ into equation 72,

$$m = \frac{dy}{dx} = \frac{3(x_P)^2 + A}{2y_P} \tag{73}$$

The new points can be found just as in the point addition by substituting $m$ into,

$$m^2 = (x_P + x_Q + x_R) \implies x_R = m^2 - x_P - x_Q \tag{74}$$

$$y^R = m(x_R - x_P) + y_P \implies -y_R = (x_P - x_R) - y_P \tag{75}$$

## 3.3 Elliptic Curves over Finite Fields

In ECC cryptography, finite fields are used.

**Definition.** A field $\mathbb{F}$ is a group of at least two elements, with two operations $+$ and $\star$, for which [7],

1. The field $\mathbb{F}$ has an identity 0 under operation $+$.
2. The field $\mathbb{F}$ has an identity 1 under operation $\star$.
3. The field $\mathbb{F}$ has distributive property: for $a, b, c \in \mathbb{F}$, $(a+b)\star c = (a\star c)+(b\star c)$.

A prime field, on the other hand, is a finite set $\mathbb{F}_p$ of modulo $p$ remainders, where $p$ is any prime number.

Elliptic curves over a finite prime field can be defined as,

$$\{(x, y) \in (\mathbb{F}_p), y^2 \equiv x^3 + ax + b (mod\, p), 4a^3 + 27b^2 \not\equiv 0 (mod\, p)\} \tag{76}$$

These elliptic curves express the same features explored previously. The only difference is the modulo $p$. To illustrate,

- Addition: $(32 + 9)(mod\, 23) = 18$
- Multiplication: $(7 \star 4)(mod\, 23) = 5$

Addition over finite field modulo $p$ is shown in Figure 9. It has the same property of addition and doubling of the elliptic curve over real numbers.

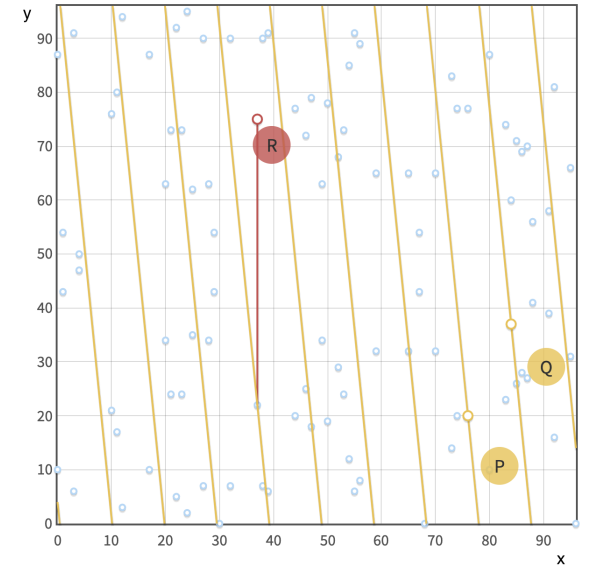



Figure 9: Addition over finite field modulo $p$.

It might seem counter-intuitive at first, but considering the shape of a finite field in Figure 10, it makes sense that lines wrap the finite field to intersect another point,

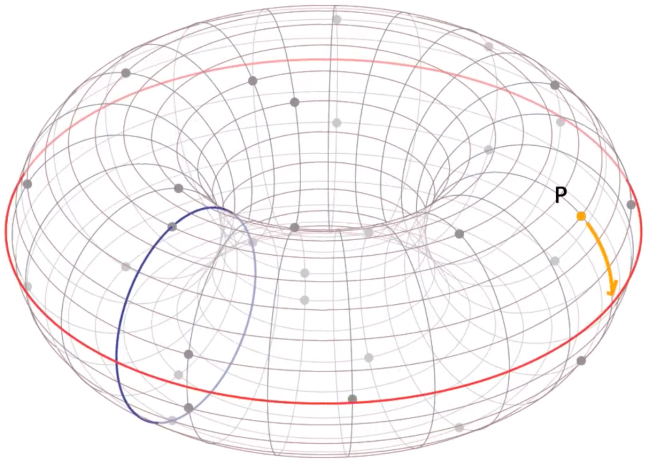



Figure 10: A finite field.

Another important property of elliptic curves over finite fields is subgroups. Since every operation is under modulo $p$, the values of the points repeat themselves after a certain point.

That is, considering the curve $y^2 \equiv x^3 + x (mod\, 13)$ and the point $G = (4, 4)$,

$$\begin{aligned} 1G &= (4,4) \quad & 6G &= (4,4) \\ 2G &= (9,6) \quad & 7G &= (9,6) \\ 3G &= (9,7) \quad & 8G &= (9,7) \\ 4G &= (4,9) \quad & 9G &= (4,9) \\ 5G &= \infty \quad & 10G &= \infty \end{aligned}$$

As can be seen the points repeats themselves in each 5 points. So, the order of the subgroup is told to be 5.

## 3.4 Fundamentals of ECDSA

ECDSA(Elliptic Curve Digital Signature Algorithm) works in a cyclic subgroup of an elliptic curve over a finite field[6]. Therefore, for ECDSA, following five parameters should be defined,

- The prime $p$ which determines the size of the finite field.
- The coefficients $a$ and $b$ of the elliptic curve equation.
- The base point $G$ which is the starting point of the subgroup.
- The order $n$ of the subgroup.
- The message $z$ that will be signed.

The public key and private key are determined as follows. The private key is a random integer $d$ chosen from the positive integers $1 \leq d \leq n-1$, where $n$ is the order of the subgroup. The public key is the point $H = dG$, where G is the base point of the subgroup. When the public key $H$ and $G$ is known, it is hard to calculate the private key $d$ due to discrete logarithm problem. However, if private key $d$ and $G$ is known, it is easy to find $H$.

In RSA, only by sending the signature, the receiver can reveal the message. However, in ECDSA, the sender reveals the message first and then releases the digital signature. This is one of the differences between RSA digital signature and ECDSA.

The ECDSA algorithm will be explored in two parts: signing and verification.

**Signing:**

For $d$ is the private key and $H$ is the public key, there are 4 steps of signing respectively,

1. Choose a random integer $k$ that is $1 \leq k \leq n-1$.
2. Calculate the point $P = kG(mod\, n)$.
3. Calculate the number r by the equivalence $r \equiv x_P(mod\, n)$ where $x_P$ is the $x$ coordinate of point $P$. If $r$ is calculated to be 0, then choose another $k$ and repeat the previous steps.
4. Calculate the number s by the equivalence $s \equiv k^{-1}(z + rd)(mod\, n)$. As defined previously, $k^{-1}$ is the expression for the modular inverse of $k$ modulo $n$. If $s$ is calculated to be 0, then choose another $k$ and repeat the previous steps.

As a result, $r$ and $s$ values are calculated. The pair of these values $(r, s)$ is the digital signature of the message $z$.

**Verification:**

In order to verify the digital signature formed by ECDSA, the public key $H$ and the digital signature $(r, s)$ will be used. There are basically three steps of verifying the signature,

1. Calculate two integers $u_1$ and $u_2$ by the following congruences,

$$u_1 \equiv s^{-1}z(mod\, n) \tag{77}$$

$$u_2 \equiv s^{-1}r(mod\, n) \tag{78}$$

2. Calculate the point $P = u_1G + u_2H(mod\, n)$.

3. If the $x$ coordinate of the point $P$ equals $r$, then the signature is valid. That is, for message $z$,

$$(r, s) \iff r \equiv x_P(mod\, n)$$

**Proof of ECDSA**

In order to prove the correctness of ECDSA, we should start from the public key $H = dG$ and the equation in the second step of verification,

$$P \equiv u_1G + u_2H(mod\, n) \tag{79}$$

$$\equiv u_1G + u_2dG(mod\, n) \tag{80}$$

$$\equiv (u_1 + u_2d)G(mod\, n) \tag{81}$$

Substitute equations 60 and 61 for $u_1$ and $u_2$,

$$P \equiv (s^{-1}z + s^{-1}rd)G(mod\, n) \tag{82}$$

$$\equiv s^{-1}(z + rd)G(mod\, n) \tag{83}$$

From the equation of $s$, $s^{-1}$ can be found,

$$s \equiv k^{-1}(z + rd)(mod\, n) \tag{84}$$

$$s^{-1} \equiv (k^{-1})^{-1}(z + rd)^{-1}(mod\, n) \tag{85}$$

Substituting equation 68 into equation 66 and $k \cdot k^{-1} \equiv 1(mod\, n)$, we reach,

$$P \equiv k(z + rd)^{-1}(z + rd)G(mod\, n) \tag{86}$$

$$\equiv k(z + rd)^{-1}(z + rd)G(mod\, n) \tag{87}$$

Since the multiplication of modular inverses is equivalent to 1 modulo n,

$$P \equiv kG(mod\, n) \tag{88}$$

Thus, we reached the second step of signing, proving ECDSA.

## 3.5 Application of ECDSA

Let's say some of the IB coordinators required verification for the finishing hour of morning examinations which $13(z = 13)$. This time IBO decides to use ECDSA.

The curve they decide on is,

$$y^2 = x^3 + x + 3 (mod\, 17) \tag{89}$$

They choose one of the points on the curve to be the generator which is point $G = (x_G, y_G) = (7, 8)$. They also decide on the public and private key: $d = 5$, $H = 5G = (3, 4)$ Using group operations, they find the order of subgroup of the curve at which point $nG = \infty$.

Let's use algebraic doubling to find $2G = (x_{2G}, y_{2G})$. First, the slope $m$ should be found,

$$m \equiv \frac{3(x_G)^2 + a}{2y_G} \equiv \frac{3(7)^2 + 1}{2 \cdot 8} \equiv \frac{37}{4} (mod\, 17) \tag{90}$$

$$\frac{1}{4} \equiv 4^{-1} \equiv 13 (mod\, 17) \tag{91}$$

$$37 \cdot \frac{1}{4} \equiv 37 \cdot 13 \equiv 481 \equiv 5 (mod\, 17) \tag{92}$$

Now, $x$ and $y$ coordinates of point $2G$ can be found,

$$x_{2G} \equiv m^2 - 2x_G = 5^2 - 2 \cdot 7 \equiv 11 (mod\, 17) \tag{93}$$

$$y_{2G} \equiv m(x_G - x_{2G}) - y_G \equiv 5(7 - 11) - 8 \equiv -28 \equiv 6 (mod\, 17) \tag{94}$$

TI-Inspire is used for calculations. Group operations can be used to find the whole subgroup,

$$1G = (7, 8) \qquad 9G = (6, 15) \qquad 17G = \infty$$
$$2G = (11, 6) \qquad 10G = (2, 8)$$
$$3G = (12, 3) \qquad 11G = (8, 9)$$
$$4G = (16, 1) \qquad 12G = (3, 13)$$
$$5G = (3, 4) \qquad 13G = (16, 16)$$
$$6G = (8, 8) \qquad 14G = (12, 14)$$
$$7G = (2, 9) \qquad 15G = (11, 11)$$
$$8G = (6, 2) \qquad 16G = (7, 9)$$

Then they follow the steps of signing,

1. They choose a random integer $k$,$1 \leq k \leq 17 - 1$: $k = 9$.
2. They calculate the point $P = 7G = (2, 9)$.
3. Then, they calculate the number r by the equivalence $r \equiv 2 (mod\, 17)$.
4. At the end, the number s is calculated as $s \equiv 7^{-1}(13 + 2 \cdot 5) \equiv 5 \cdot 23 \equiv 13 (mod\, 17)$.

As IBO created the signature for its message $z = 13$, they share the signature $(r, s)$ along with their public key $H = (3, 4)$ and generator point $G = (7, 8)$.

IB coordinators validate the signature by the following steps,

1. They calculate two integers $u_1$ and $u_2$ by the following congruences,

$$u_1 \equiv 13^{-1}13 \equiv 4 \cdot 13 \equiv 1 (mod\, 17) \tag{95}$$

$$u_2 \equiv 13^{-1}2 \equiv 4 \cdot 2 \equiv 8 (mod\, 17) \tag{96}$$

2. Then, they calculate the point $P = u_1G + u_2H (mod\, 17) = 1G + 8H$. To calculate it, IB coordinators use point addition and doubling. However, since we know the private key $d$, let's calculate the result easily. Since $H = 5G$,

$$P = 1G + 8 \cdot 5G = 41G \tag{97}$$

   Since this is a cyclic group with an order of n subgroup, we can take modulo 17,

$$P \equiv 7G \equiv 41G (mod\, 17) \tag{98}$$

$$P = (2, 9) \tag{99}$$

3. They know check if the signature is valid by using the following formula,

$$r \equiv x_P (mod\, n)$$

$$2 \equiv 2 (mod\, 17)$$

Thus, the finishing time of morning exams are digitally signed by IBO and validated by IB coordinators by ECDSA.

# 4 Conclusion

In this paper, RSA digital signature and ECDSA are explored through their underlying mathematics and two example of their real-world application. RSA depends on the arithmetic of prime numbers, whereas ECC depends on the arithmetic of elliptic curves over finite fields. The properties of prime numbers and elliptic curves results in a function that is easy to calculate with a key but very difficult to calculate without one. Both of the algorithms are currently popular and used extensively. However, both of them have their own strengths. For example, where RSA is a simpler method to implements than ECC, ECDSA offers better protection at larger key lengths as shown in Figure 11.

| Security Bit Level | RSA | ECC |
|---|---|---|
| 80 | 1024 | 160 |
| 112 | 2048 | 224 |
| 128 | 3072 | 256 |
| 192 | 7680 | 384 |
| 256 | 15360 | 512 |

Figure 11: Comparison of RSA and ECC key lengths for the same security level[8].

Internet of things refers to devices that can interact with the environment and also connected to the internet. Since it is connected to the internet, it requires a secure channel to communicate information. In the following decades, we will probably see that devices will become smaller as internet of things devices such as wearable health devices, smart home security systems, and nanobots become part of our lives. On the other hand, the computer power will increase more which might cause problems making easier to hack a device. Therefore, in a small device against a powerful computer, better efficiency in security level/ key length will be required, which means that ECC will become more popular in the near future.

It should also be noted that there is a growing threat to these digital signatures due to the possible power of quantum computer which can break the security of the cryptography applications as they are much faster than conventional computers. While there are currently post-quantum schemes developed for ECC[5], the same can not told for RSA. As a result, as the need for security level increases due to stronger computers and the possibility of quantum computer, ECC will likely be favored for its higher performance(security level: key length ratio) at higher security levels.

# References


[1] W. Diffie and M. Hellman. "New directions in cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[2] R.L. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the ACM* 21 (1978), pp. 120–126.

[3] Kenneth H. Rosen. *Elementary Number Theory and its Applications*. 6th Edition. Pearson/Addison-Wesley, 2005.

[4] Tilborg Henk C A van. *Fundamentals of cryptology a professional reference and interactive tutorial*. Kluwer Academic Publishers, 2005.

[5] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. *Post-Quantum Cryptography*. Springer Nature, 2009.

[6] Darrel R. Hankerson, Scott A. Vanstone, and Alfred J. Menezes. *Guide to elliptic curve cryptography*. Springer, 2011.

[7] Diane Herrmann and Paul Sally. *Number, shape, and symmetry: An introduction to number theory, geometry, and group theory*. CRC Press, 2013.

[8] Dindayal Mahto and DILIP YADAV. "RSA and ECC: A comparative analysis". In: *International Journal of Applied Engineering Research* 12 (Jan. 2017), pp. 9053–9061.

[9] Youssef El Housni. "Introduction to the Mathematical Foundations of Elliptic Curve Cryptography". In: *HAL Open Science* (2018).