2017

Elliptic Curve Cryptography and Quantum Computing

Emily Alderson
Ouachita Baptist University

ELLIPTIC CURVE CRYPTOGRAPHY AND QUANTUM COMPUTING

AN HONORS THESIS BY:

EMILY ALDERSON

TABLE OF CONTENTS

PREFACE

In the year 2007, a slightly nerdy girl fell in love with all things math. Even though she only was exposed to a small part of the immense field of mathematics, she knew that math would always have a place in her heart. Ten years later, that passion for math is still burning inside. She never thought she would be interested in anything other than strictly mathematics. However, she discovered a love for computer science her sophomore year of college. Now, she is graduating college with a double major in both mathematics and computer science.

This nerdy girl is me. It was my first semester, freshman year at Ouachita Baptist University when I discovered an interest in cryptography. Discrete Mathematics had a small portion of the class that discussed various types of cryptosystems. These were the simplified versions that were easier to comprehend so that we could truly grasp the underlying concepts of cryptography. However, I craved more. I wanted to delve further into the advanced world of cryptography. It was not until my Honors Directed Study that I was able to do so.

During my directed study, I was able to research the history of cryptography and what it has transformed into today. While it was fascinating to look at all the various cryptosystems that have been created throughout time, one cryptosystem caught my eye in particular—Elliptic Curve Cryptography, commonly referred to as ECC. I had heard of a few cryptosystems but never the ECC. As such, my interest was piqued. Having the opportunity to research Elliptic Curve Cryptography, I was ready to try and understand a new mathematical concept. I was ready to further expand my love and passion for mathematics, and I was not disappointed. Learning even the basics of elliptical curves proved challenging. The math behind ECC is immense and quite frankly, intimidating. Nevertheless, I do not back down from a challenge…at least not a mathematical one.

My research of elliptic curves, the basis of Elliptic Curve Cryptography, opened up my eyes to an entirely new field of mathematics. The challenges that arose during my studies, while frustrating at times, are well worth the wealth of knowledge I was able to acquire. The research in my study of elliptical curve cryptography only increased the ten-year passion for mathematics that is still inside the slightly nerdy girl.

This paper is the culmination of all my research over elliptic curves. It reflects the knowledge that I was able to acquire while studying elliptic curve cryptography and quantum computers.

THE HISTORY

Cryptography has grown immensely since the beginning of time. Each society throughout history has shown incredible skill and thought to keep its secrets safe: from the early scytale, to the Atbash cipher, to the Caesar cipher, to the modern cryptosystems of today such as the Rivest-Shamir-Adleman encryption and Elliptic Curve Cryptography. Each of these has been a way for a society to protect information deemed important. There has always been a chase between the code makers and the code breakers. Each new achievement in technology allows the cryptanalysts, the code breakers, to find a way to break the current cryptosystem. Nevertheless, those same advancements in technology also allow for the cryptographers, the code makers, to create a new and more secure cryptosystem.

Because each technological advance brings the threat of breaking the current cryptosystem, cryptographers constantly try to create a new cryptosystem that is better than the previous. The standard of a cryptosystem was held by the Rivest-Shamir-Adleman encryption method, commonly referred to as RSA encryption. RSA encryption is in fact still widely used in today's society. Elliptic Curve Cryptography (ECC) developed as an alternative to RSA encryption. The idea of using elliptic curves for a new type of cryptosystem first appeared in 1985, when Neal Koblitz and Victor Miller proposed the idea ("Elliptic curve cryptography"). Most intriguing though, is that Koblitz and Miller proposed their idea independently, without knowledge of the other proposing the exact same idea. After the initial proposition of using elliptic curves as a cryptosystem in 1985, ECC did not gain popularity until the late 1990's. At this point, a majority of companies standardized on ECC, so it started to receive commercial acceptance. Today, ECC is mainly used in resource constrained environments such as ad-hoc wireless networks and mobile networks since a large key size is not required.

The Elliptic Curve Cryptosystem was designed and created as a high-level security public key cryptography system. A public key cryptography system includes a pair of mathematically related keys: one public that is distributed widely and one private that only the owner knows (GlobalSign). To encrypt a message, the public key is used. To decrypt a message, the private key is used. Thus, the public/private key pair has been found to be a safe and effective way to keep data secure since the private key is only known to the owner. Conventional public key cryptosystems typically result in lower speeds and an increase in the consumption of bandwidth because of the large size of the key to maintain the required level of security. Elliptic curves decrease the key size needed to maintain a similar level of security.

Cryptosystems, such as RSA encryption, depend on very large prime numbers. These large numbers require more bits to store as a key. ECC does not require such large numbers to create a secure key. To understand how an Elliptic Curve Cryptosystem achieves a high level of security with a compact size, a basic understanding of elliptic curves is needed.

A BASIS OF ELLIPTIC CURVES

An elliptic curve is simply a type of cubic curve which has an order of three. Another way to state this is that at least one of the variables in the equation is raised to the third power. However, an elliptic curve is not a function. For a function, each input has a single output, but an output may have more than one input. However, an elliptic curve may have two outputs for every one input thus making it not a function. We can see this in Figure 1.

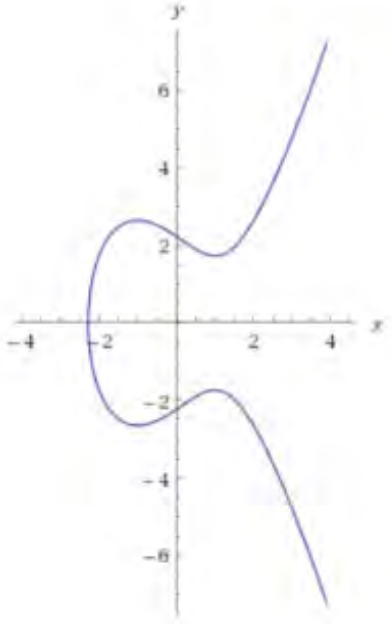

Figure 1. A sample of an elliptic curve

An elliptic curve is a nonsingular cubic curve in two variables over either an infinite or finite field. Typically, we concentrate on elliptic curves over a finite field that is algebraically closed, meaning every non-constant polynomial has a root. As such, a nonsingular cubic curve has nine points of inflection, although only three of these may be real. A point of inflection is a point on a curve in which the second derivative of an equation switches from positive to negative or vice versa (WolframAlpha). It

is where the curve changes from concave up to concave down or from concave down to concave up. An elliptic curve uses the Weiertstrass equation:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_5 \quad (1)$$

such that $a_1, a_2, a_3, a_4, a_5 \in \mathbb{R}$ and $\Delta \neq 0$, in which Δ is the discriminant of Equation 1. Equation 1 can be simplified to:

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

in which $a_1, a_2, a_3 = 0$ and $a_4 = a$ and $a_5 = b$

Or

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

in which $a_1, a_2 = 1$ and $a_3, a_4 = 0$ and $a_5 = b$. The solutions of Equations (1), (2), and (3) are confined to a region of space that is topologically equivalent to that of a torus, which resembles something along the lines of a doughnut as we can see in Figure 2.

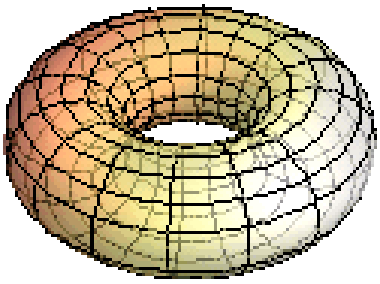

Figure 2. A common 3 dimensional torus

Several properties of elliptic curves allow them to form an Abelian group under addition. An Abelian group G:

1) Has closure – for all a, b in G, $a * b$ is also in G
2) Has associativity – for all a, b, and c in G, the equation $(a * b) * c = a * (b * c)$ holds true
3) Has an identity – there exists an element e in G such that for all elements a in G, the equation $e * a = a * e$ holds true
4) Has an inverse – For each a in G, there exists an element a^{-1} in G such that

   $a * a^{-1} = a^{-1} * a = e$

5) Has commutativity – for all a, b in G, the equation $a * b = b * a$ holds true

It is because of these properties that addition and point doubling on elliptic curves are not too difficult. To add two points, P and Q, on an elliptic curve, a line from P to Q intersects a point $(-R)$ on the elliptic curve. Then, the reflection of $(-R)$ about the x – axis is taken to find R, which is $R = P + Q$ (Md. Al-Amin Khandaker Nipu). Formally, we say:

Let $P = (x_P, y_P)$ and $Q = (x_Q, y_Q)$ be elements of E (an elliptic curve) with $P \neq Q$.

Then, $R = P + Q = (x_R, y_R)$ such that:

$$x_R = \lambda^2 - x_P - x_Q \quad (4)$$

$$y_R = \lambda(x_P - x_R) - y_P \quad (5)$$

Where $\lambda = \frac{y_Q - y_P}{x_Q - x_P}$

Graphically, we see point addition in Figure 3 on the elliptic curve $y^2 = x^3 - 3x + 5$ as:

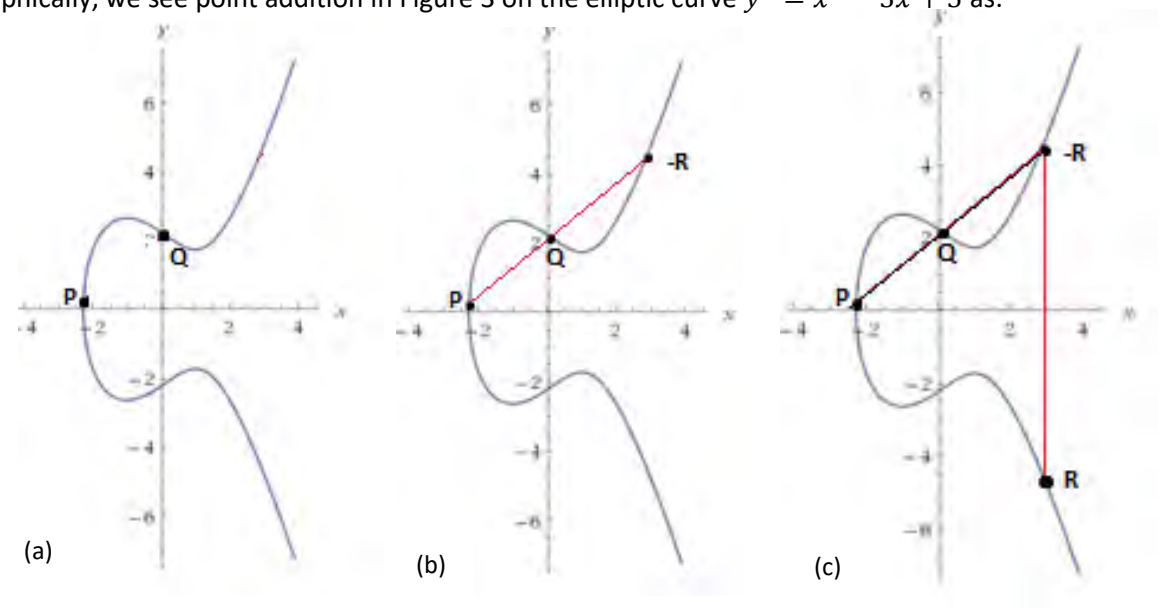

Figure 3. (a) Two points on an elliptic curve. (b) Line from point P through point Q to intersect at point –R. (c) R = P + Q is found by taking the reflection of –R over the x – axis.

To perform point doubling on an elliptic curve, we take the tangent line of the point we are doubling, P, and find where it intersects the elliptic curve, $(-Q)$. Then, we take the reflection about the x – axis of this point to find our doubled point, Q (Md. Al-Amin Khandaker Nipu). Formally, we say:

Let $P = (x_P, y_P)$ be an element of E with $P \neq -P$. Then $R = 2P = (x_R, y_R)$ such that

$$x_R = \lambda^2 - 2x_P \quad (6)$$

$$y_R = \lambda(x_P - x_R) - y_P \quad (7)$$

Where $\lambda = \frac{3x_P^2 + a}{2y_P}$

Graphically, we see point doubling in Figure 4 on the elliptic curve $y^2 = x^3 - 3x + 5$ as:

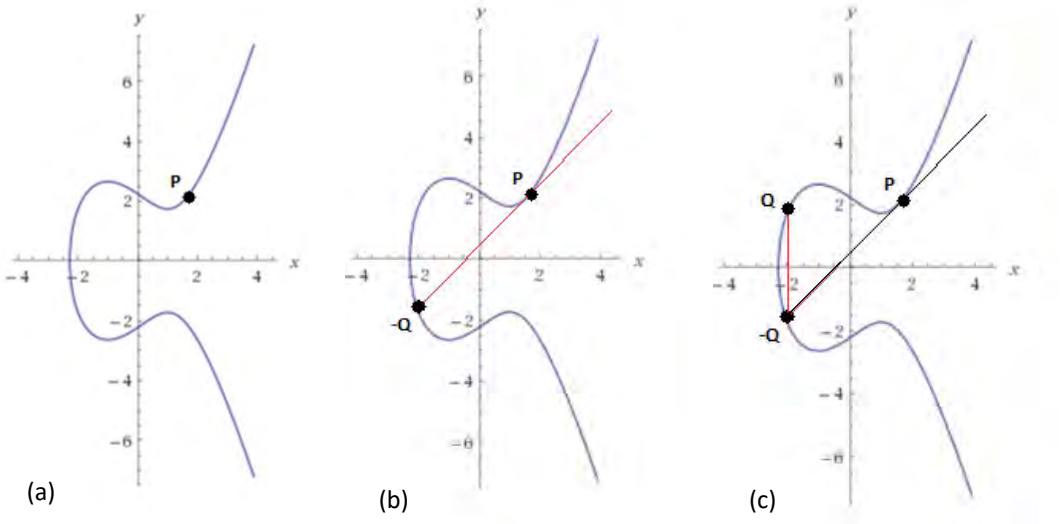

Figure 4. (a) Point P on an elliptic curve. (b) Tangent line at P to find –Q. (c) Q = 2P is found by taking the reflection of –Q across the x – axis

The examples of point addition and point doubling above are of elliptic curves over the real numbers. While elliptic curves over the real numbers are easy to demonstrate and easy to look at graphically, they are not the best option. Elliptic curves over the real numbers have slower calculations and there are inaccuracies due to rounding errors. There is also the fact that the real numbers are an infinite field. Elliptic curves over a prime field or over a binary field are much more effective in calculations. A prime field is a finite field in which every element is a prime number. A binary field is a

finite field in which every element is a binary number. An 8-bit binary number is eight digits comprised of either a 1 or a 0. For example, the number 6 in binary is represented as 00000110. It is much more difficult to interpret the graphs of prime and binary fields than a graph of the real numbers. Elliptic curves over a prime field are formally defined as:

$$y^2 \bmod p = (x^3 + ax + b) \bmod p \quad (8)$$

Where $(4a^3 + 27b^2) \bmod p \neq 0$ and $x, y, a, b \in [0, p-1]$ (McGivern). When using point addition or point doubling on an elliptic curve over a prime field, modular arithmetic of addition, subtraction, multiplication, and inversion must be performed. The graph of an elliptic curve over a prime field is shown in Figure 5.

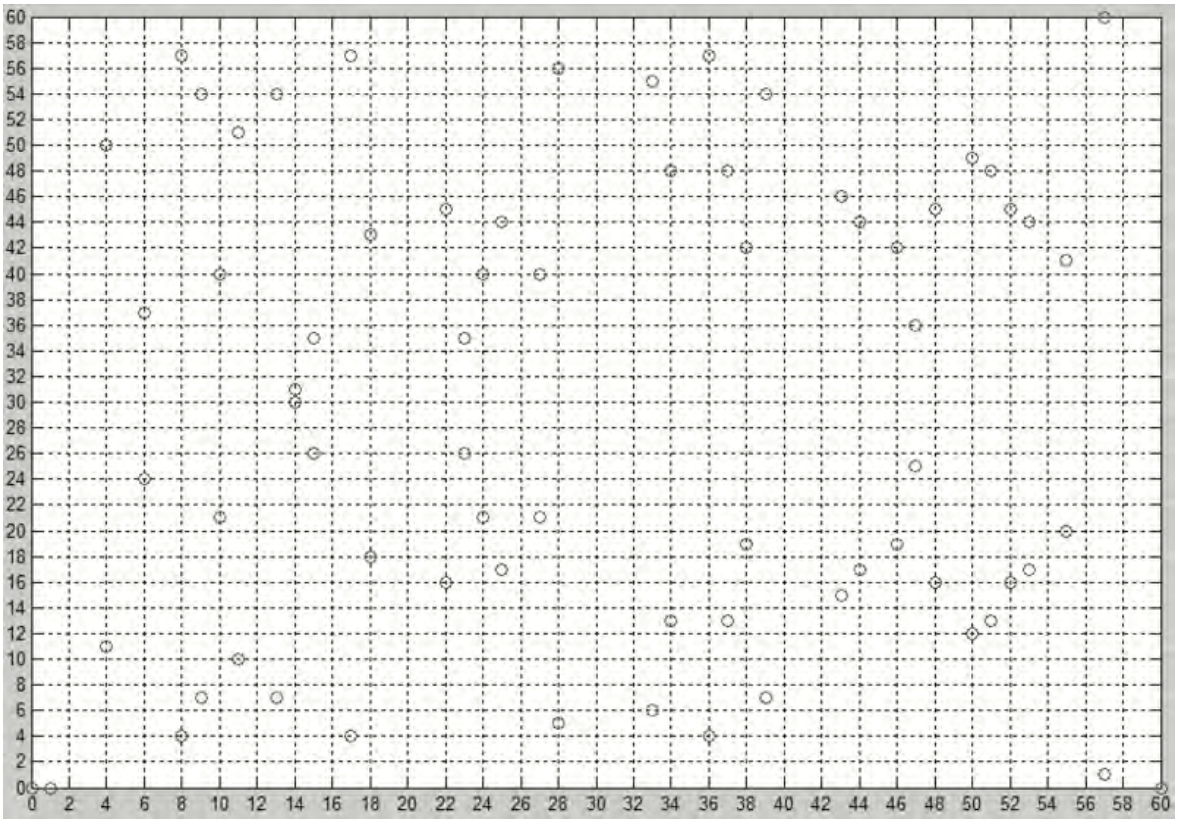

Figure 5. The graph of an elliptic curve over a primary field.

Elliptic curves over a binary field are defined as:

$$y^2 + xy = x^3 + ax^2 + b \tag{9}$$

where $x, y, a, b \in \mathbb{F}_{2^m}$ such that $m \in \mathbb{Z}^+$ (Bluhm). Thus, we assume that the finite binary field has an irreducible polynomial and a single element is a generator for the entire field. The field is generated by taking the powers of that single element. The graph of an elliptic curve over a binary field is as follows.

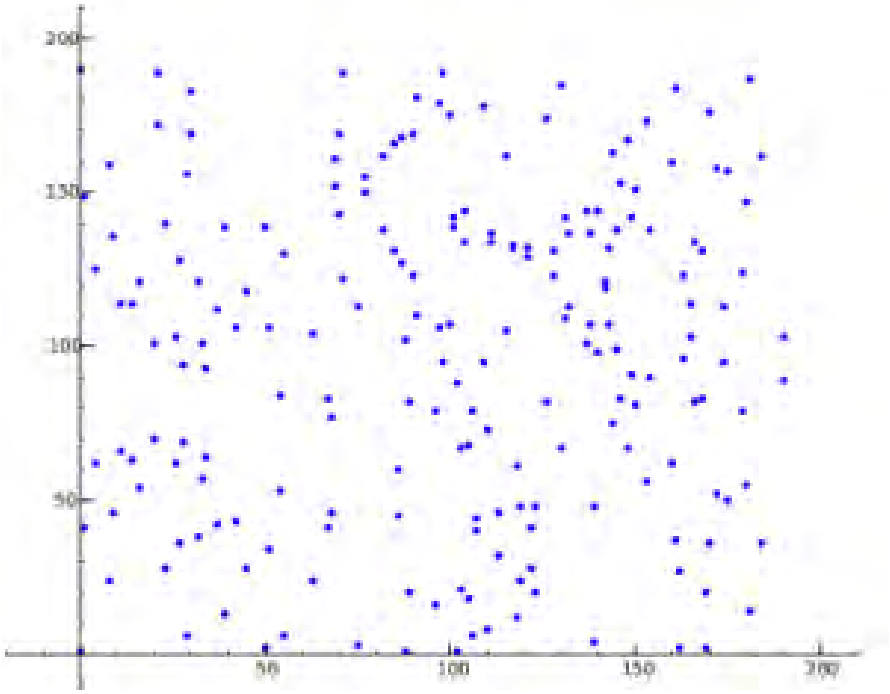

Figure 6. The graph of an elliptic curve over a primary field.

Normal point representation on an elliptic curve is an affine coordinate (x, y). With this normal pair coordinate, point doubling and point addition are at a disadvantage. Inverse operations are involved with both point doubling and point addition which can get very expensive in terms of computation time. Instead, the normal Cartesian points can be represented as a triplet (X, Y, Z). These triplets are called projective coordinates. The relationship between the Cartesian points (x, y) and the triplets (X, Y, Z) is:

$$(X, Y, Z) = \left(\lambda^c x,\ \lambda^d y\ , \lambda\right) \tag{10}$$

Where $\lambda \neq 0$ and

$$(x, y) = \left(\frac{X}{Z^c}, \frac{Y}{Z^d}\right) \tag{11}$$

The use of projective coordinates instead of Cartesian coordinates can avoid the use of inverse operations and because inverse operations are so time costly, projective coordinates save time. The minor drawback is that projective coordinates do require more multiplications in the field operation.

Elliptic curves over prime fields differ from elliptic curves over binary fields as each is better suited for different applications. An elliptic curve over a prime field is best for software applications. As such, the prime field does not require the extended "bit-fiddling" operations required by those elliptic curves over a binary field. However, an elliptic curve over a binary field is better suited for hardware applications. The binary field can allow us the opportunity to take less logic gates to create a cryptosystem compared to those elliptic curves over a prime field. Whether elliptic curves are over a prime field or a binary field, they can be used to construct a secure public key cryptography system. A private d is randomly selected from $[1, n-1]$ such that $n \in \mathbb{Z}^+$. The public key Q is computed by $d * P$ in which P and Q are points on the elliptic curve. This is denoted as a scalar multiplication and can also be used for the signature, encryption, and key agreement in an ECC system.

ELLIPTIC CURVE CRYPTOSYSTEM

An elliptic curve cryptosystem is a type of public key cryptosystem. Every public key cryptosystem is constructed on the basis of the complexity of one or more difficult mathematics problems. The Rivest-Shamir-Adleman encryption method uses the math problem of factoring. Another algorithm, the Diffie-Hellman encryption method, uses the math problem of a discrete logarithm. Elliptic Curve Cryptography is built on the basis of the elliptic curve discrete logarithm problem (The Elliptic Curve Discrete Logarithm Problem). The problem is:

> Suppose E is an elliptic curve over $\frac{\mathbb{Z}}{p\mathbb{Z}}$ and $P \in E(\mathbb{Z}/p\mathbb{Z})$. Given multiple Q of P, the elliptic curve discrete logarithm problem is to find $n \in \mathbb{Z}$ such that $nP = Q$.

It is because of the difficulty of this problem that ECC has a smaller key size than that of the RSA encryption method. The elliptic curve discrete logarithm problem does not require the sub-exponential time algorithm it takes to solve the math problems for both RSA and Diffie-Hellman. Because it only takes a sub-exponential time to solve the problems for RSA and DH, the key sizes must be larger. We see that for the same *n* length, the sub-exponential running time is less than the exponential running time.

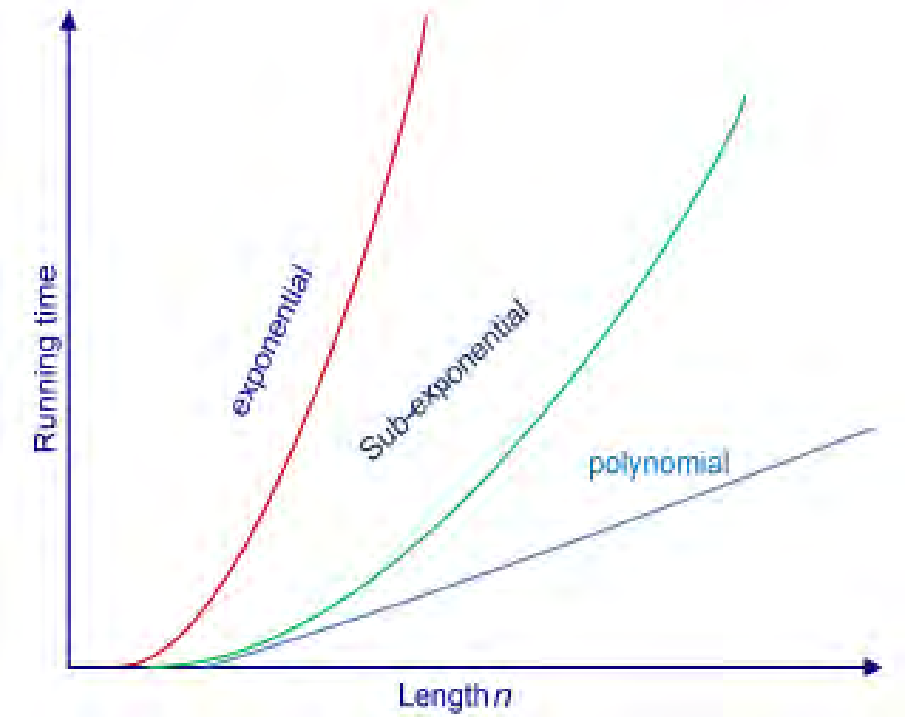

Figure 7. Graph that shows how different algorithms' (exponential, sub-exponential, and polynomial) running times increase as the length *n* increases

Thus, to have the same level of security as an exponential running time algorithm, a sub-exponential running time algorithm must have a larger key size—or length n. Because of the smaller key, the elliptic curve cryptosystem does not nearly take up the same amount of space as the RSA encryption method for the same level of security. Table 1 compares the number of bits needed for RSA and Diffie-Hellman versus that of ECC to maintain the required level of security in differing sizes of symmetric algorithms.

Symmetric Algorithm (bit)	RSA and DH (bit)	ECC (bit)
56	512	112
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 1. Shows for differing values of a symmetric algorithm the amount of bits that is needed to maintain a high level of security for both RSA and ECC.

To keep the same level of security, RSA and DH encryption require much larger key sizes than the ECC. In today's society, 15,360 bits is computationally infeasible in embedded systems. Thus, the Elliptic Curve Cryptosystem, at 521 bits, might be the future. It should be noted that an embedded system is different from the general purpose computer. A general purpose computer is designed to manage a wide range of processing tasks while an embedded system is designed to perform only a certain task (Techopedia). As such, design engineers of embedded systems can optimize cost, power consumption and size. Examples of embedded systems include MP3 players, digital watches, switches, giant routers, among many others.

Elliptic Curve Cryptosystems also have a digital signature. This works in a manner similar to the keys. Again, because of the way ECC works, the size needed to create a digital signature is much less than that of RSA or Diffie-Hellman. Typically, to create a digital signature for a piece of data, a cryptographic hash of the data is created with a mathematical operation with the private key. This is

the hash value or a checksum value (Sullivan). If the data has been modified, the value will be different. Because of this, the integrity of the data can be evaluated (Techopedia).

Some benefits to using ECC over RSA for a cryptosystem are listed below.

Elliptic Curve Cryptosystem (ECC)	**Rivest-Shamir-Adlman (RSA)**
Smaller public and private keys	Public keys are six times larger and private keys are 12 times larger (at a 128-bit level)
Public key signature validation is slower than RSA	Public key signature validation is generally faster than ECC
Private key operations (examples: signature generation, key management) are faster	Private key operation are ten times slower (at a 128-bit level) and increase to be fifty or one hundred times slower (at a 256-bit level of security)
The algorithm used is an exponential algorithm	The algorithm used is a sub-exponential algorithm
Low on CPU consumption	Higher on CPU consumption
Low on memory usage	Higher on memory usage

Table 2. Compares ECC to RSA, showing the benefits of using ECC.

Elliptic Curve Cryptosystems are becoming more popular because they are faster than their counterpart in the field of cryptography. This does not mean they are safer or even less safe. They are just different in the mathematical problem used to maintain security. There have been some concerns growing about the effectiveness of ECC encryption.

The main problem with the Elliptic Curve Cryptosystem is the effectiveness of the private key. Choosing a private key is pivotal in creating a secure system. This step cannot be overlooked or the cryptosystem will be rendered useless. One case of this happening is in 2010 when Sony used a static d in their private key $Q = d * P$. Because d was static instead of random, a group called *failOverflow* was able to recover the private key. In 2011, another problem with ECC was found but then quickly remedied. Two researchers showed it was possible to retrieve the private key of a server using OpenSSL that uses ECC over a binary field by implementing a timing attack ("Elliptic Curve Digital Signature Algorithm"). A timing attack in cryptography is when the attacker analyzes the time taken to execute cryptographic algorithms in order to compromise the system. Using this, the attacker may be able to backtrack to the input itself. However, this is not always a feasible approach because executions may take average time performance or the worst-case performance time. The vulnerability the two researchers found was fixed with OpenSSL 1.0.0e. In August 2013, another issue announced itself with Elliptic Curve Cryptography. The Java class SecureRandom had some bugs in its implementations that sometimes generated collisions in the d value of the private key $Q = d * P$. However, this problem can be prevented by the deterministic generation of d ("Elliptic Curve Digital Signature Algorithm"). This guarantees true randomness of d.

For the past 20 years, the National Security Agency (NSA) has been promoting ECC as a more secure alternative to RSA; however, in August 2015, the NSA stopped promoting for the deployment of Elliptic Curve Cryptography. This caused speculations that Elliptic Curve Cryptography is not truly secure. Many theorize that the reason the NSA stopped promoting Elliptic Curve Cryptography is because it is viewed only as a stopgap solution for the looming threat of quantum computers. Some others still believe the NSA stopped promoting ECC because of some drawback of the encryption system. Nevertheless, quantum computing is a real threat to the cryptographic community.

QUANTUM COMPUTING

Quantum computers are a thing of the future and that future might be more near than we think. However, true quantum computing would render all current cryptosystems useless. The speeds at which quantum computers could execute algorithms would make the difficult mathematical problems cryptosystems rely on very feasible.

A traditional computer uses bits which can be represented as a 1 (on) or a 0 (off). A quantum computer uses qubits—quantum bits. A quantum system with qubits encodes the 0s and 1s into two distinguishable quantum states. Thus qubits can use superposition and entanglement because qubits behave "quantumly" (University of Waterloo). Superposition is the ability of a quantum system to be in multiple states at the same time—such as "up" and "down." Thus, a qubit can be both "on" and "off" at the exact same time. Entanglement is an extremely strong correlation between quantum particles. This correlation is so strong that two or more particles can be linked in perfect unison even across unimaginable distances.

Because of superposition and entanglement, a quantum computer is able to process a large number of calculations simultaneously. The bits of the computer are both "on" and "off," allowing for much faster computations. This means that quantum computers could factor very large numbers in practically no time at all. This would render RSA encryption pointless. Not only would quantum computers make RSA completely moot, but they would also make every current cryptosystem pointless with the amazing computations quantum computers could theoretically accomplish.

D-Wave: The Quantum Computing Company was founded in 1999 and is the world's first quantum computing company, as well as being the leader in the development of quantum computing software and systems ("D-Wave Systems"). For years, quantum computers have just been research,

theory, and proposals. D-Wave is one of the companies that are making quantum computers a reality. No longer is quantum computing just theory; D-Wave is implementing quantum computing in the 2000 qubit D-Wave 2000Q quantum computer. It is the most advanced quantum computer in existence and is based on a novel type of superconducting processor that uses quantum mechanics to accelerate computations (D-Wave Systems Inc.). With quantum computing, D-Wave quantum computers have a large advantage over conventional computers. In fact last year, a team of NASA and Google scientists found that a D-Wave quantum computer was 100 million times faster than a conventional computer (Beall). Even the computing giants Microsoft and Google are working on forms of quantum computing (Nature Journal). It is a feasible possibility that quantum computers could be something of the norm in the coming years.

If this is the case, we must consider what it will take to design a cryptosystem against quantum computers. It might be that we must have quantum computing to design a quantum-state encryption method. Then, the race will be on between the hackers and the cryptographers. Will the cryptographers be able to create a cryptosystem that withstands quantum computing before the hackers crack all current encryption methods? This is the question that we must keep in mind in the coming years.

CONCLUSION

I am thankful for the opportunity to research elliptic curve cryptography and quantum computers. I was able to experience a field of mathematics I had not previously been exposed to in a classroom setting. Writing this paper expanded my knowledge and introduced me to mathematical proofs written at an extremely high caliber.

However, being exposed to such high level proofs forced me to examine them at a slow pace to make sure that I thoroughly understand not only the math but also the concepts behind the math. There were several times when going through a proof that I would have to search other theorems or lemmas to help me better understand the original proof. This cost me quite a bit of time and slowed down my progress in understanding elliptic curve cryptography. I found that elliptic curves are a fascinating subject area with many interesting characteristics that allow them to be used for cryptography. Originally I did not know that when used in cryptography, elliptic curves are used in either a prime field or a binary field. This severely affects how the point doubling and point addition works with elliptic curves and makes the mathematics much more complicated.

When researching quantum computing, I was fascinated how much progress had been made. A few years ago, quantum computers were just theory and "what ifs." Now, prototypes are being created and quantum computers are a real possibility. They are no longer just an idea and concept. While this is exciting, it is also very frightening. Quantum computers could render all cryptosystems useless. We now have to create a new kind of cryptography that would be able to withstand the computational power of a quantum computer.

I loved having the opportunity to research both elliptic curve cryptography and quantum computers to further my knowledge on those subjects.

WORKS CITED

Beall, Abigail. "Inside the weird world of quantum computers." WIRED UK. WIRED UK, 23 Mar. 2017. Web. 28 Mar. 2017. <http://www.wired.co.uk/article/quantum-computing-explained>.

Bluhm, Manuel. "Software optimization of binary elliptic curves arithmetic using modern processor architectures." Department of Mathematics, University of Haifa, 17 June 2013. Web. 2 Feb. 2017. <https://www.emsec.rub.de/media/attachments/files/2014/11/MA_Bluhm.pdf>.

"D-Wave Systems." Meet D-Wave | D-Wave Systems. N.p., n.d. Web. 10 Apr. 2017. <https://www.dwavesys.com/our-company/meet-d-wave>.

D-Wave Systems Inc. "D-Wave Systems." Quantum Computing | D-Wave Systems. N.p., n.d. Web. 8 Jan. 2017. <https://www.dwavesys.com/quantum-computing>.

"Elliptic curve cryptography." Wikipedia. Wikimedia Foundation, 29 Mar. 2017. Web. 30 Mar. 2017. <https://en.wikipedia.org/wiki/Elliptic_curve_cryptography>.

"Elliptic Curve Digital Signature Algorithm." Wikipedia. Wikimedia Foundation, n.d. Web. 27 Jan. 2017. < https://en.wikipedia.org/wiki/Elliptic_Curve_Digital_Signature_Algorithm>

GlobalSign. "What is Public-key Cryptography?" What is an SSL Certificate. N.p., n.d. Web. 7 Dec. 2016. <https://www.globalsign.com/en/ssl-information-center/what-is-public-key-cryptography/>.

McGivern, James. "ECC vs RSA: Battle of the Crypto-Ninjas." LinkedIn SlideShare. N.p., 11 July 2014. Web. 2 Mar. 2017. <https://www.slideshare.net/JamesMcGivern/ecc-vs-rsa-battle-of-the-cryptoninjas>.

Md. Al-Amin Khandaker Nipu, PhD Student, Secure Wireless System Lab Follow. "Elliptic Curves and Elliptic Curve Cryptography." LinkedIn SlideShare. N.p., 19 May 2016. Web. 8 Feb. 2017. <https://www.slideshare.net/eNipu/elliptic-curves-and-elliptic-curve-cryptography>.

Nature Journal. "Quantum computers ready to leap out of the lab in 2017." Nature News. Nature Publishing Group, n.d. Web. 27 Mar. 2017. <http://www.nature.com/news/quantum-computers-ready-to-leap-out-of-the-lab-in-2017-1.21239>.

Sullivan, Nick. "ECDSA: The digital signature algorithm of a better internet." Cloudflare Blog. Cloudflare Blog, 13 May 2015. Web. 19 Apr. 2017. <https://blog.cloudflare.com/ecdsa-the-digital-signature-algorithm-of-a-better-internet/>.

Techopedia. "What is a Cryptographic Hash Function? - Definition from Techopedia." Techopedia.com. N.p., n.d. Web. 10 Mar. 2017. <https://www.techopedia.com/definition/27410/cryptographic-hash-function>.

Techopedia. "What is an Embedded System? - Definition from Techopedia." Techopedia.com. N.p., n.d. Web. 3 Feb. 2017. <https://www.techopedia.com/definition/3636/embedded-system>.

The Elliptic Curve Discrete Logarithm Problem. N.p., n.d. Web. 13 Jan. 2017. <http://wstein.org/edu/2007/spring/ent/ent-html/node89.html>.

University of Waterloo. "Quantum computing 101." Institute for Quantum Computing. N.p., 11 Nov. 2013. Web. 18 Mar. 2017. <https://uwaterloo.ca/institute-for-quantum-computing/quantum-computing-101>.

WolframAlpha. "Inflection Point." Inflection Point -- from Wolfram MathWorld. N.p., n.d. Web. 30 Jan. 2017. <http://mathworld.wolfram.com/InflectionPoint.html>.

WolframAlpha. Wolfram|Alpha: Computational Knowledge Engine. N.p., n.d. Web. 2 Mar. 2017. <https://www.wolframalpha.com/>.