The Digraph of the Square Mapping on Elliptic Curves

Katrina Glaeser

September 19, 2009

Abstract

Consider a subgroup of an elliptic curve generated by a point P of order n. It is possible to match any point Q to an integer $k \pmod{n}$ such that $Q = kP$ using a brute force method. By observing patterns in the digraph of the squaring map on the integers modulo n it is possible to perform this matching. These techniques can be applied to solving the Elliptic Curve Discrete Log Problem given a complete graph of the square mapping $k \cdot P \to k^2 \cdot P$ for the elliptic curve points.

1 Introduction

The structure of the mapping $kP \to k^2P$ in elliptic curves, is identical to the squaring map on the group $\mathbb{Z}_n$ where n is the order of the point P. The elements of this group are represented by coordinates belonging to the chosen elliptic curve E. The point addition operation is well understood and efficient to perform, similar to addition in $(\mathbb{Z}_n, +)$. The scalar multiplication operation $(k, P) \to kP$ for $k \in \mathbb{Z}_n$ and $P \in E$ which corresponds to the multiplication $(k, a) \to k \cdot a \pmod{n}$ in the group $\mathbb{Z}_n^*$ is easy to perform. However, the inverse of scalar multiplication $(kP, P) \to k$ is much harder than $(ka, a) \to k \pmod{n}$ in $\mathbb{Z}_n^*$, which can be computed by finding $a^{-1} \pmod{n}$. This combination of a simple scalar multiplication function, and a significantly more difficult inverse function which makes elliptic curves well suited for use in cryptography.

The study of cryptography involves both creating cryptographic schemes, and testing their resilience to various types of attack. If there is some way for an hacker to repeatedly encrypt a chosen plaintext and obtain the resulting cyphertext, some information about the encryption system being used might be compromised. Assuming it is possible to obtain a graph of the function $kP \to k^2P$ using a repeated chosen plaintext attack, it is also possible to obtain some information about the map $kP \to k$ by observing similarities between the squaring map over elliptic curves, and the squaring map in $\mathbb{Z}_n$.

The elliptic curve Diffie Hellman problem can be solved with a solution to the the elliptic curve discrete log problem, but not vice versa. The elliptic curve squaring oracle makes it possible to solve the elliptic curve Diffie Hellman problem efficiently. Given the digraph of the elliptic curve squaring map, the elliptic curve discrete log problem can be reduced to the problem of finding an isomorphism to the digraph of the squaring map that preserves addition and sends 1 to P. It is possible to find the elliptic curve discrete log of a point by observing patterns in the squaring map, but to be useful, it must also be more efficient than the methods for finding the elliptic curve discrete log that do not rely on an understanding of the elliptic curve squaring map.

2 Preliminaries

2.1 Oracles

An **oracle** is a theoretical machine that can give an output to a specific problem in a single operation. The **squaring oracle** for an elliptic curve refers to an oracle that is able to compute k^2P given the point kP.

2.2 Functional Graphs

A **functional graph** of a function f is a directed graph (**digraph**) associated with a function. Each node represents an element of the domain and there will be a directed edge from $x \to y$ if and only if $f(x) = y$.

A **cycle** occurs when there is a path from a node to itself. A **t-cycle** occurs in a graph of a function $x \to f(x)$ when $f^t(x) = x$ but $f^c(x) \neq x$ for any integer $0 < c < t$. The nodes in the t-cycle are the nodes $x, f(x), \ldots, f^{t-1}(x)$ and these nodes are called **cyclic**. Cycles of length one, are often called **fixed points**. Notice that in a functional graph the nodes need not all be cyclic.

The **indegree** of a node is the number of directed edges leading into a node. Similarly, the **outdegree** of a node is the number of outbound edges. The outdegree of every node in a functional graph will be always be one. A **leaf** is a node with indegree zero.

A **path** in a graph is a sequence of nodes with an edge between each consecutive nodes. Two nodes are **connected** if there exists a path between them. A **component** containing x is the set of all the nodes connected to x. A **tree** is a graph where each pair of nodes is connected by exactly one path. A **binary tree** is a tree where each node has indegree two or zero. A **complete binary tree** is a binary tree where each leaf has the same depth. Let A and B be sets, and let $f : A \to A$ and $g : B \to B$ be functions. Let $h : A \times B \to A \times B$ be defined $h(a, b) = (f(a), g(b))$. Then the functional graph of h on the set $A \times B$ is the **cross product** of the graph of f on A with the graph of g on B.

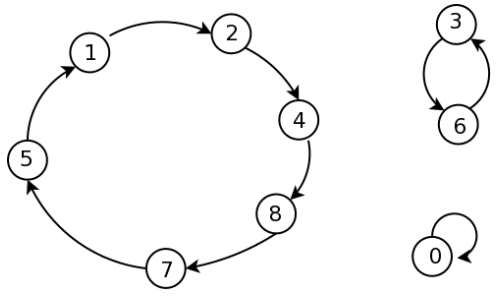

Figure 1: Graph of $x \to 2x \pmod 9$

The **squaring map modulo** n is the function $x \to x^2 \pmod n$. The **elliptic curve squaring map** is the map $kP \to k^2P$. The **doubling map modulo** n is the function $x \to 2x \pmod n$.

2.3 Elliptic Curves

An **elliptic curve** E is the set of points satisfying the equation

$$E : y^2 = x^3 + ax + b \tag{1}$$

Notice that a graph of this curve will be symmetric across the x-axis. This equation can be considered over any field, but for this paper we will consider E over the the finite field $K = \mathbb{Z}_p$ for a prime p.

$$E(K) := y^2 = x^3 + ax + b \pmod{p} \tag{2}$$

There is a group law defined on the points of an elliptic curve, which can be described in three cases. In order to add two distinct elliptic curve points P and Q draw a line through the two points. In the

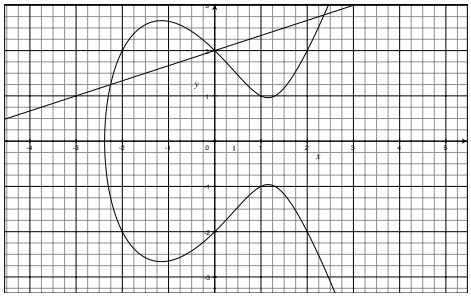

Figure 2: The curve $y^2 = x^3 - 4x + 4$ with a line intersecting at three points

first case, this line will intersect the curve at a third point R. Reflect the point $-R$ across the x-axis (as shown); the reflected point R is the sum of P and Q. We write $P \oplus Q = R$. In order to add a point P to

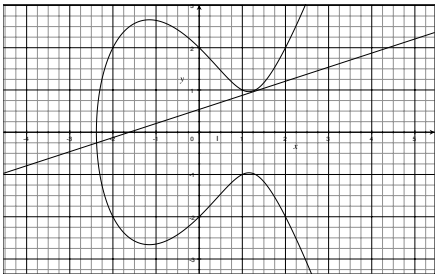

Figure 3: The curve $y^2 = x^3 - 4x + 4$ with a tangent line intersecting the curve twice

itself, draw a line tangent to the curve at P. The tangent line should intersect the curve at another point. As in the previous case, we label this point R. We can reflect R across the x-axis to obtain $-R$. We say that $P \oplus P = -R$ or that $2 \cdot P = -R$.

We must also consider the case where the line through P and Q or the tangent line at P does not intersect the curve. When this happens we say that that the $P \oplus Q = \mathbf{0}$ or that $2 \cdot P = \mathbf{0}$. This $\mathbf{0}$ is the additive identity for the group law on E.

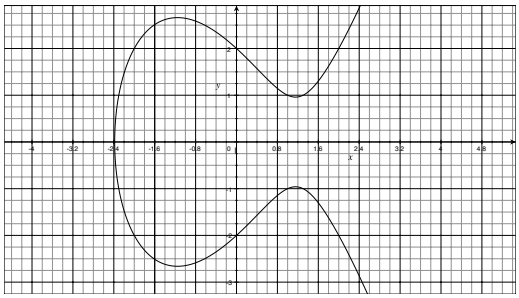

Figure 4: The curve $y^2 = x^3 - 4x + 4$ with a vertical line intersecting P and $-P$

2.4 The Elliptic Curve Discrete Log Problem

The **Elliptic Curve Discrete Log Problem** is the problem of finding the integer k given the points $Q = kP$ and P on an elliptic curve E. The integer k is called the elliptic curve discrete log of Q with respect to P.

The multiple of a point kP is relatively easy to calculate, taking $O(\log k)$ elliptic curve point addition and doubling operations using the double and add method of point multiplication [4]. In comparison, a brute force method of finding the integer k given the points $Q = kP$ and P, would take $O(k)$ elliptic curve point addition operations. Collision methods can be used to find the elliptic curve discrete log in $O(\sqrt{p})$ elliptic curve point additions. The Weil pairing can be used to imbed an elliptic curve E defined over $\mathbb{F}_p$ into an extension $\mathbb{F}_{p^j}$, which reduces the problem to finding the discrete log in ${\mathbb{F}_{p^j}}^*$ [4] which provides improved efficiency when k is small.

2.5 The Elliptic Curve Diffie Hellman Problem

The **Elliptic Curve Diffie Hellman Problem** is the problem of finding the point abP given the points aP, bP and P on an elliptic curve E. The security of elliptic curve Diffie Hellman key exchange and the elliptic curve ElGamal cryptosystem depend on the difficulty of this problem (see [4]).

In the presence of a elliptic curve squaring oracle, the elliptic curve Diffie Hellman problem can be efficiently solved. Given aP and bP it is possible to compute $(a+b)P$ efficiently. The elliptic curve squaring squaring oracle can be used to find the points a^2P, b^2P and $(a+b)^2P$. The points $(-a^2)P$ and $(-b^2)P$ can be found by reflection across the x-axis.

$$\begin{aligned} (a+b)^2P &= (a^2+2ab+b^2)P \\ (a^2+2ab+b^2)P \oplus -a^2P \oplus -b^2P &= 2abP \end{aligned}$$

The point abP is a point $(\bar{x}, \bar{y}) \in E$ where the tangent line to E at $(\bar{x}, \bar{y})$ intersects E at $2abP$. Then abP can be calculated by computing the equation of the tangent line to the curve E at any point, and solving for the point of intersection with the point $2abP$.

The elliptic curve Diffie Hellman problem can be reduced to the problem finding elliptic curve discrete log of either aP or bP, but there is no know reduction in the other direction. This motivates the question: what can be determined about the elliptic curve discrete log operation from a squaring map on an elliptic curve?

3 Structure in the Squaring Map

3.1 Complete Graphs for Prime n

Let us examine the functional graph mapping $x \to x^2 \pmod{n}$ where n is prime.

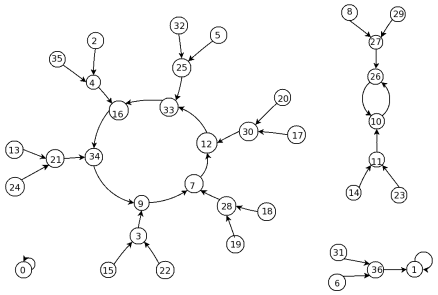

Figure 5: Graph of $x \to x^2 \pmod{37}$

Notice that if $n-1$ is factored as $n-1 = 2^h \cdot q$ with q odd, then we will find that there are q nonzero cyclic nodes. Each cyclic node is the root of a binary tree of height h [1]. This occurs because the nonzero elements of $\mathbb{Z}_n$ coincide with the elements of the multiplicative groups ${\mathbb{Z}_n}^*$ when n is prime. From the existence of primitive roots [3], we know that ${\mathbb{Z}_n}^* \cong \mathbb{Z}_{n-1}$ using the discrete log map $x \to \log_g(x)$ where g is a generator. Under the discrete log, the map $x \to x^2$ becomes $log_g(x) \to \log_g(x^2)$ which is equivalent to the map $log_g(x) \to 2\log_g(x)$. This means that the squaring map on ${\mathbb{Z}_n}^*$ looks like the doubling map $x \to 2x$ in $\mathbb{Z}_{n-1}$.

Since $\gcd(2^h, q) = 1$, by the Chinese Remainder Theorem $\mathbb{Z}_{n-1} \cong \mathbb{Z}_{2^h} \times \mathbb{Z}_q$. The doubling map on $\mathbb{Z}_{n-1}$ can be represented as the cross product of the graphs of the doubling maps on $\mathbb{Z}_{2^h}$ and $\mathbb{Z}_q$.

Proposition 1. *For prime n, 1 is the only non-zero fixed point for the squaring map modulo n.*

Proof. Let n be prime. Fixed points in the squaring map modulo n will occur only at nodes x only when

$$\begin{aligned} x^2 &\equiv x \pmod{n} \\ x^2 - x &\equiv 0 \pmod{n} \\ x(x-1) &\equiv 0 \pmod{n}. \end{aligned}$$

Since n is prime this means that either $x \equiv 0 \pmod{n}$ or $x - 1 \equiv 0 \pmod{n}$. Hence 0 and 1 are the only fixed points. □

Proposition 2. *For prime $n > 2$ each non-zero node will have indegree two or zero in the squaring map.*

Proof. If a node a is a quadratic nonresidue then a has indegree 0. Otherwise let $x^2 \equiv a \pmod{n}$, then there is an edge from x to a. Since $(-x)^2 \equiv x^2 \pmod{n}$ there must also be an edge from $-x$ to a. Since $n > 2$ is prime and $a \neq 0$ so $x \neq -x$. Hence if there is one edge into a then there are at least 2 edges into a. Suppose there was also an edge from a node y to a, then

$$\begin{aligned} x^2 &\equiv y^2 \pmod{n} \\ x^2 - y^2 &\equiv 0 \pmod{n} \\ (x+y)(x-y) &\equiv 0 \pmod{n} \end{aligned}$$

So either $y \equiv x$ or $y \equiv -x$. Therefore there are no more that 2 edges into any node a. □

3.2 Complete Graphs for Composite n

Proposition 3. *For prime a, if $a|n$ and $a|x$ then $a|y$ for any y connected to x.*

Proof. Suppose $n = {p_1}^{\alpha_1} \cdot {p_2}^{\alpha_2} \cdot \ldots \cdot {p_r}^{\alpha_r}$. First we show that for a node corresponding to the integer x, if we have that $p_i|x$ then the node corresponding to $x^2 \pmod n$ will also be divisible by p_i, and conversely, $p_i|x^2$ implies that $p_i|x$ for $1 \leq i \leq r$.

Suppose that $p_i|x$ for $1 \leq i \leq r$. If $y \equiv x^2 \pmod n$ then $y = x^2 + t \cdot n$ for some integer t. $p_i|x$ implies $p_i|x^2$ and $p_i|n$ implies $p_i|tn$. This means $p_i|x^2 + tn$, and hence $p_i|y$.

Suppose $p_i|y$ for $1 \leq i \leq r$. If $y \equiv x^2 \pmod n$ then $x^2 = y + s \cdot n$ for some integer s. Since $p_i|y$ and $p_i|n$ we know that $p_i|sn$. By adding the terms together we have $p_i|(y + sn)$ which means $p_i|x^2$. If $p_i|x^2$ then $p_i|x$.

This means that if there is an edge between two points x and y, and x is not relatively prime to n, then the neither is y. If two x and y elements are in the same component, then this property applies to each of the edges along the path connecting the two nodes. So if x is not relatively prime to n i.e. $p_i|x$, then the neither is y because $p_i|y$. □

Proposition 4. *The nodes in ${\mathbb{Z}_n}^*$ will be in disjoint components from the nodes which are not relatively prime to n.*

Proof. Let $x \in {\mathbb{Z}_n}^*$. Then $\gcd(x, n) = 1$ so there is no prime p such that $p|n$ and $p|x$. Suppose that x is connected to y and there exists prime p where $p|n$ and $p|y$, then by (Prop. 3) $p|x$, contradiction. Therefore, if a node x is not relatively prime to n and x is in the same component as y, then y is also not relatively prime to n. Similarly, if x is relatively prime to n, and x is in the same component as y then y must also be relatively prime to □

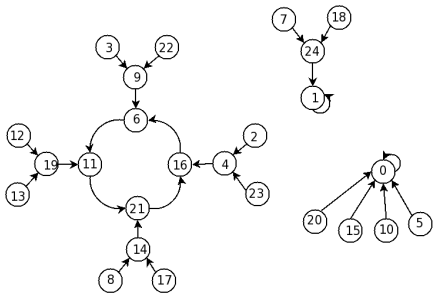

Figure 6: Graph of $x \to x^2 \pmod{25}$

Let us examine the functional graph mapping $x \to x^2 \pmod n$ where n is composite, but where there exists a primitive root in $\mathbb{Z}_n$.

Notice that if $\phi(n)$ is factored as $\phi(n) = 2^h q$, where ϕ is the Euler phi function, we will find that when examining just the elements of ${\mathbb{Z}_n}^*$, there are q cyclic nodes . Each of the q cyclic nodes is the root of a binary tree of height h [1].

When ${\mathbb{Z}_n}^*$ does not have primitive roots, the patterns observed differ slightly. The nodes that belong to

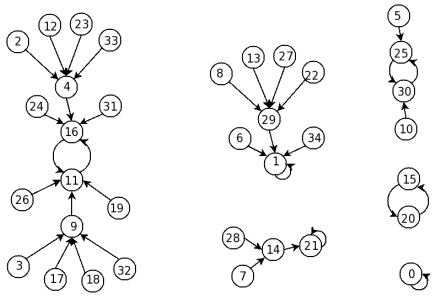

Figure 7: Graph of $x \to x^2 \pmod{35}$

${\mathbb{Z}_n}^*$ follow the same pattern where for $\phi(n) = 2^h \cdot q$ for odd q, there are q cyclic nodes, but the non-cyclic nodes leading to the cyclic nodes are no longer necessarily in binary trees. This happens because by the fundamental theorem of finitely generated abelian groups, ${\mathbb{Z}_n}^*$ is isomorphic to the direct product of cyclic groups of prime power order. The multiplication operation in ${\mathbb{Z}_n}^*$ is isomorphic to the addition operation in ${\mathbb{Z}_{{p_1}^{a_1}}}^+ \times \ldots \times {\mathbb{Z}_{{p_r}^{a_r}}}^+$, so as a result the squaring operation $x \to x \cdot x$ is isomorphic to the doubling operation $x \to x + x$ in ${\mathbb{Z}_{{p_1}^{a_1}}}^+ \times \ldots \times {\mathbb{Z}_{{p_r}^{a_r}}}^+$.

Consider the figure $x \to x^2 \pmod{35}$. $\phi(35) = 24 = 8 \cdot 3$. There are 24 elements relatively prime to 35, and of those elements, there are three elements in cycles. Each cyclic element is the root of a tree with eight elements. The tree of eight elements is not a binary tree. This is because ${\mathbb{Z}_{35}}^* \cong \mathbb{Z}_3 \times \mathbb{Z}_4 \times \mathbb{Z}_2$. The doubling map in $\mathbb{Z}_4 \times \mathbb{Z}_2$ does not create a binary tree.

Many more results about the distribution of cycle lengths and the patterns have been discussed in [1], [5] and [8].

4 Results

4.1 The Squaring Map for Elliptic Curves

Consider an cyclic subgroup of an elliptic curve E generated by the point P. If P generates n points including $\mathbf{0}$. Then the digraph of the map $kP \to k^2P$ will be isomorphic to the digraph of the map $x \to x^2 \pmod{n}$, under the elliptic curve discrete log function. The isomorphic binary trees attached to the cyclic nodes $Q \in \{kP | k \in \mathbb{Z}_n^*\}$ create components with radial symmetry. Given a known pair (k, kP) some information can be determined about other points in the same component by examining the position of a point relative to a point whose elliptic curve discrete log is known.

4.2 Strategies for Matching Points to Integers

Given the elliptic curve discrete log of a point kP, it is possible to identify the point corresponding to $-kP$ because when k is relatively prime to n, kP and $-kP$ are always the only two nodes with edges to k^2P. Regardless of the information from the squaring map, the negative of a point is easy to compute because the negative of a point corresponds to the reflection across the x-axis. Using only modular arithmetic operations it is also possible to calculate $k^2 \pmod{n}$ and match it to the point k^2P. Since it is possible to both match

negatives, and calculate modular squares of integers without using elliptic curve point operations, given a pair (k, kP) it is possible to label $k^{2^j} \cdot P$ and $(-k^{2^j}) \cdot P$ for $j \in \mathbb{Z}_{\geq 0}$ with their corresponding integers modulo n using only modular arithmetic operations, by repeatedly squaring.

Notice that the sequence of points $k^{2^j} \cdot P$ and $(-k^{2^j}) \cdot P$ will eventually repeat once the cyclic nodes have been reached. This implies that any that the elliptic curve discrete log of any cyclic point not connected to $\mathbf{0}$ (or it's negative) can be matched using only modular arithmetic operations given the discrete log of any point in the same component.

If a point kP is a cyclic node in a component that is large relative to the size of n, then the probability that a point in the same component can be matched by computing random scalar multiples of points, is high. The average number of modular arithmetic operations needed to reach a specified cyclic node from a known point is dependent on h and the average cycle lengths in the doubling map modulo q. For n with few cyclic nodes and small cycles this could mean a significant reduction in the average number of modular squaring operations needed to find the elliptic curve discrete log of the cyclic nodes

In the doubling map modulo q, t-cycles can occur only when there are integers s where $2^t s \equiv s \pmod{q}$. Let Q be in a component with a t-cycle with $Q = kP$ for some k with $\gcd(k, n) = 1$. Let g be a primitive root for ${\mathbb{Z}_n}^*$ where n is prime. By calculating and testing only points $(g^{s2^h})P$ where $2^t s \equiv s \pmod{q}$ the points $(g^{s2^h})P$ will all be in cycles with cycle lengths that divide t. This can reduce the number of points to be tested in order to find a node in the same component as Q, particularly if there are very few cycles with lengths that divide t.

For prime n, a modification of Shank's algorithm for taking square roots modulo n can compute square roots with $O(\log(q) + h^{\frac{3}{2}})$ modular arithmetic operations for $n = 2^h q + 1$ [7]. This means that if a node Q in the elliptic curve squaring map is matched to its elliptic curve discrete log k, then any nodes with edges to Q can also be matched using this square rooting algorithm. This algorithm can be used to discover one of the square roots. In the case where n is prime, if a square root exists, there are two square roots (by Prop. 2). Given $\sqrt{k} \pmod{n}$, computing the point $\sqrt{k}P$ will allow you to match $\sqrt{k} \pmod{n}$ to $\sqrt{k}P$ and by process of elimination $-\sqrt{k} \pmod{n}$ to $-\sqrt{k}P$.

When considering prime n the component containing $\mathbf{0}$ is an isolated fixed point, but but for composite n it may not be isolated. For composite $n = {p_1}^{\alpha_1} \cdot {p_2}^{\alpha_2} \cdot \ldots \cdot {p_r}^{\alpha_r}$ with at least one prime factor, nodes kP with k divisible by at least one of each prime factor $p_1, p_2, \ldots p_r$, will be connected to the $\mathbf{0}$ node. There are several representations of elliptic curve points [2], but the zero point should be distinctive and therefore easy to detect. If a point Q is known to be in the same component as $\mathbf{0}$ the elliptic curve discrete log of Q must be divisible by all of the prime divisors of n.

For n with a primitive root, if the point $Q = kP$ is in a component that does not have the tree structure predicted from the factorization of $\phi(n) = 2^h q$, then the elliptic curve discrete log of Q must not be relatively prime to n, which eliminates several possible k. While searching the graph data for patterns in the tree structure may be cumbersome with a complete graph, it may be an efficient attack when only a partial graph of the elliptic curve squaring map can be obtained.

4.3 Efficiency Analysis

In the previous section several methods were described for matching elliptic curve points to their discrete log based in various cases. In order to determine the cases where these methods are most efficient, the run-time bounds must be given. By examining the parameters of the run-time bounds for each method, the cases where the method is most advantageous can be discovered.

(Several methods have been evaluated in Table 1.)

Table 1: Methods

Operation	Algorithm and Run-time Bounds	Advantages and Strategies
Locating the fixed points to find the point P in the squaring map.	Finding fixed points in an adjacency matrix, or adjacency list will take $O(n)$ operations to each point to see if it has an edge to itself.	This can be avoided if the fixed points are flagged while the squaring map is being created, or if the generator P is public.
Creating a list of the nodes in a connected component containing Q, where there are s elements in the component	A depth first search can be used to create a list of nodes which are connected to Q. A depth first search takes $O(\|E\|)$ steps to perform, where E is the set of edges. This is a functional graph, so the number of edges is equal to the number of nodes this will take $O(s)$ operations.	This is most efficient when s is small relative to n i.e. when the component containing Q is small.
Finding the elliptic curve discrete log of any point in a component of size s given a list of nodes in a component	By repeated point addition, a collision with some point in the component will occur after an average of $O(\frac{n}{s})$ elliptic curve point addition operations.	Collisions are more likely when the component containing Q is large, i.e. when s is large compared to n.
Find a square root of x modulo n for prime n	Where $n = 2^h q$ finding a square root will require $O(\log(q) + h^{\frac{3}{2}})$ operations. (See [7])	This method of modular square rooting is most efficient when q is large with respect to n. (Recall that $\log_2(2^h q) = h + \log_2(q)$.)
After taking a modular square root, given $\sqrt{x}$ and $-\sqrt{x}$, Q and $-Q$, determine which integer corresponds to which point.	By performing elliptic curve scalar multiplications, calculate $\sqrt{x}P$ which will take $O(\log n)$ elliptic curve operations, to determine whether $\sqrt{x}P = Q$ or $-Q$.	Since this must be performed after each square root, the nodes closest to the cycle require the fewest square rooting operations and will be the most efficient to calculate.

5 Conclusion and Future Work

The squaring map on an elliptic curve can be used to understand the scalar multiplication operation and its inverse the elliptic discrete log. If the point kP is easy to match to the integer $k \pmod n$ then the existence of a squaring oracle could compromise the security of the integer k. There are some choices of n and k that make the point kP easily identifiable. The memory required to store this graph, and the cost of detecting cycles and other structures makes working with the squaring oracle cumbersome.

In practice, it may not be possible to generate a complete graph using a repeated chosen plain text attack. This could mean that not all of the points of the elliptic curve generated by P are known. An incomplete graph could also be a graph where all of the points are known, but many of the edges have not been discovered. The complete graph may not be necessary to find the Elliptic Curve Discrete Log of a point if the component on the squaring map is complete, or if some conditions apply. Future work can be done in determining what information can be extracted from various incomplete graphs.

References

[1] Blanton E., Hurd, S. P., McCranie, J. S., "On the Digraph Defined by Squaring Mod m, When m Has Primitive Roots" *Congressus Numerantium* Vol.82, pp. 167-177, 1991.

[2] Cohen, H., Miyaji, A., Ono, T., "Elliptic curve exponentiation using mixed coordinates" *Lecture Notes in Computer Science* Vol.1514, pp. 51-65, 1998.

[3] Hungerford, T.W., *Abstract Algebra: An Introduction* Thompson Learning, USA, 1997.

[4] Koblitz, N., *A Course in Number Theory and Cryptography* Springer-Verlag New York, Inc., New York, NY, 1994.

[5] Lucheta, C., Miller, E., Reiter, C., "Digraphs from powers modulo p" *Fibonacci Quarterly* Vol.34 I.3, pp. 226-239, 1996.

[6] Rogers, T. D., "The graph of the square mapping on the prime fields" *Discrete Mathematics* Vol.148, pp. 317-324, 1996.

[7] Schlage-Puchta, J., "On Shank's algorithm for modular square roots" *Appl. Math. E-Notes* Vol.5, pp. 84-88, 2005.

[8] Somer, L., Krizek, M., "Structure the digraph associated with quadratic congruences with composite moduli" *Discrete Mathematics* Vol.306, pp. 2174-2185, 2006.