Trial multiplication is not optimal but...
On the symmetry of finite cyclic groups $(\mathbb{Z}/p\mathbb{Z})^*$

Antonio Sanso
antonio.sanso@gmail.com

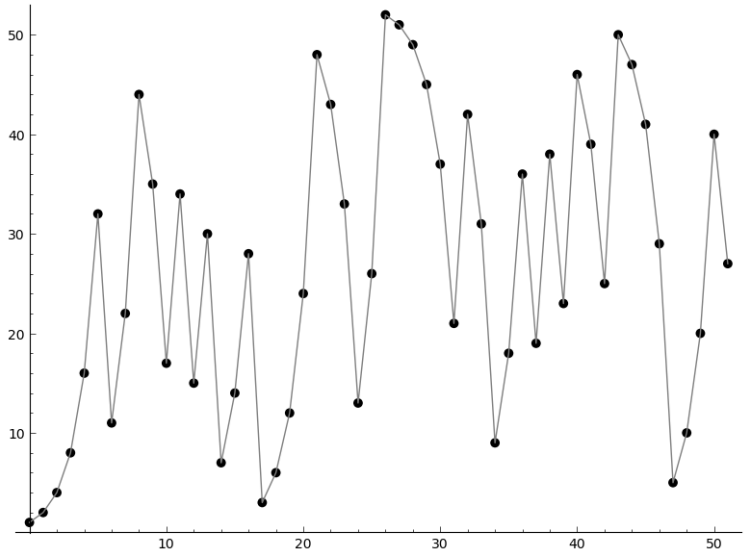
Abstract

The Discrete Logarithm Problem is at the base of the famous Diffie Hellman key agreement algorithm and many others. The key idea behind Diffie Helmann is the usage of the Discrete Logarithm function in $(\mathbb{Z}/p\mathbb{Z})^*$ as a trap door function. The Discrete Logarithm function output in $(\mathbb{Z}/p\mathbb{Z})^*$ seems to escape to any attempt of finding some sort of pattern. Nevertheless some new characterization will be introduced together with a novel and more efficient trial multiplication algorithm.

The Discrete Logarithm Problem

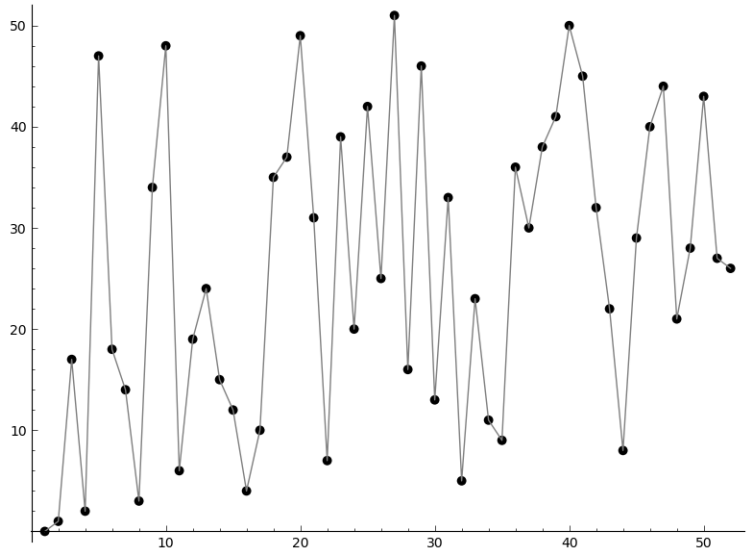
The Discrete Logarithm Problem (DLP) is at the base of many cryptographic techniques. This paper will focus only on DLP for finite cyclic group G in $(\mathbb{Z}/p\mathbb{Z})^*$ (prime moduli) of order $p-1$ with generator g. Having a finite cyclic group G and a generator g, the DLP of α, denoted $\text{dlog}_g\alpha$, is the unique integer x, $0 \leq x \leq (p-1)$, such that $\alpha = g^x$. The finite cyclic group can also be denoted by G = $\{g_0,g_1,...,g_{p-2}\}$ where $g_i = g^i \pmod p$ for $0 \leq i \leq p-2$. Using as example g = 2 and p = 53 we can try to plot G = {1, 2, 4, 8, 16, 32, 11, 22, 44, 35, 17,....,27} having Cartesian coordinates $x_i = i$ and y_i=g_i for $0 \leq i \leq (p-2)$. This draws the plot:

Figure 1: $x_i = i$ and y_i=g_i for $0 \leq i \leq (p-2)$



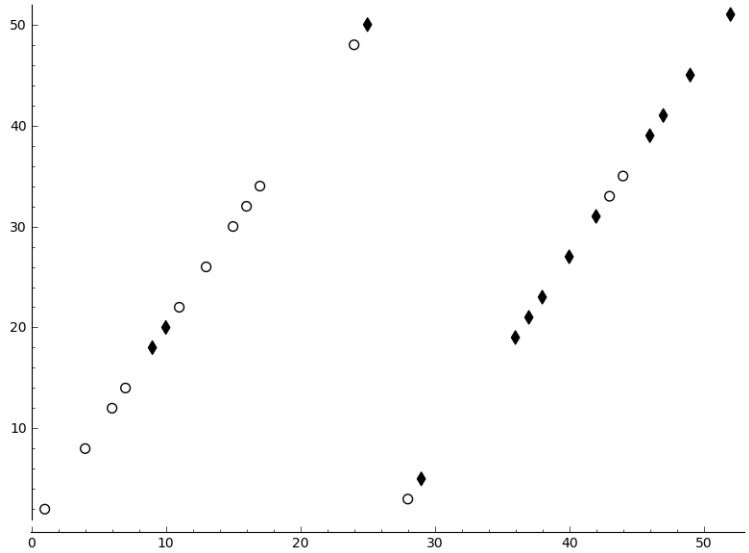
Another way to plot is to have $x_i = i$ and y_i=dlog$_g$i for $0 \leq i \leq (p-2)$.

Figure 2: $x_i = i$ and y_i=dlog$_g$i for $0 \leq i \leq (p-2)$



As shown in Figure 1 and Figure 2 the plots seem to bounce at random around the numbers. Some kind of order seems to emerge though if we calculate the Cartesian coordinate as follow: $x_i = g_i$ and $y_i = g_{(i+1)}$ $0 \leq i \leq (p-1)/2$.

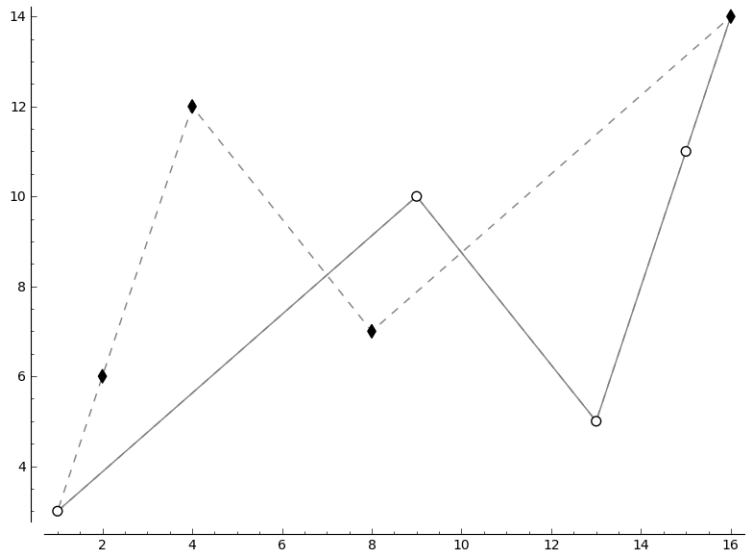
Figure 3: $x_i = g_i$ and $y_i = g_{(i+1)}$ $0 \leq i \leq (p-1)/2$, p = 53 and g = 2



In the Figure 3 the ○ indicate the first $(p-1)/4$ coordinates while the ♦ indicate the last $(p-1)/4$ coordinates.

This sort of symmetry is maybe more clear for $p = 17$ and g = 3, see below:

Figure 4: $x_i = g_i$ and $y_i = g_{(i+1)}$ $0 \leq i \leq (p-1)/2$, p = 17 and g = 3



As per Figure 3 also in the Figure 4 the ○ indicate the first $(p-1)/4$ coordinates while the ◆ indicate the last $(p-1)/4$ coordinates. Moreover the line (—) indicates the connections between the first $(p-1)/4$ coordinates and the dotted line (- - -) indicates the connections between the last $(p-1)/4$ coordinates. This kind of structured order is as well present for bigger p and bigger g but it is just harder to see plotted in a graphic. Hence this visual symmetry yells that some kind of formula should exists on the cyclic group G. This is indeed represented by:

Proposition 1.1 (finite cyclic groups symmetry)**.** *For* $1 \leq i \leq (p-1)/2$ *,*

$$g_{i+(p-1)/2} = g_{i-1+(p-1)/2} - g_i + g_{i-1}$$

Proof. By Fermat's Little Theorem we can derive that given a finite cyclic group G in $(\mathbb{Z}/p\mathbb{Z})^*$ (prime moduli) we have

$$g_{((p-1)/2)} = -1 = -g_{(-1)} + g_{(-1)} - 1 = g_{((p-1)/2-1)} + g_{(-1)} - 1$$

multiplying both sides with g^i the proposition follows. □

Example 1.2. As an example for $p = 53$, $g = 2$ and $i = 1$ we have

$$g_{1+(53-1)/2} = g_{1-1+(53-1)/2} - g_1 + g_{1-1}$$

hence

$$g_{27} = g_{26} - g_1 + g_0 = 52 - 2 + 1 = 51$$

indeed

$$2^{27} \pmod{53} = 51$$

Trial multiplication algorithm

The trivial way to compute $\text{dlog}_g\alpha$ is using brute force (this algorithm is called *trial multiplication* or *exhaustive search*). This obvious algorithm looks like:

Algorithm 1.3 (Calculate Discrete Log)**.** Compute g^0,g^1,g^2... until α is obtained

1. [Initialize] Set i =0
2. [Power] Set $a = g^i \pmod{p}$.
3. [Step equals ?] If a is equal to α output i and terminate .
4. [Try next] Set i =$i + 1$ and go to Step 2

This algorithm is clearly inefficient for large value of p and it requires $O(p)$ multiplications and more efficient algorithms exist (e.g. *number field sieve*, *index calculus*, etc.)

We can try to leverage the Proposition 1.1 in order to see if we can obtain a more efficient version of the *trial multiplication* algorithm.

Algorithm 1.4 (Calculate Discrete Log using symmetry)**.** Compute g^0,g^1,g^2... until α is obtained leveraging the Proposition 1.1

1. [Initialize] Set $i = 0$, $previous = 0$, $b = n$.
2. [Power] Set $a = g^i \pmod{p}$.
3. [Step equals ?] If a is equal to α output i and terminate .
4. [symmetry] Set $b = b - a + previous$.
5. [Symmetry equals ?] if b is equal to α output $i + (p-1)/2$ and terminate .
6. [Try next] Set $i = i + 1$, $previous = a$ and go to Step 2

Algorithm 1.4 is still inefficient compared to the *number field sieve*, *index calculus* algorithms but it is more efficient than the *naive trial multiplication* algorithm. Indeed Algorithm 1.4 requirs $O(p/2)$ multiplications and $O(2 * p/2)$ additions.

Conclusions

A new Proposition on finite cyclic groups has been proposed together with an improved trial multiplication algorithm to solve the Discrete Logarithm Problem. The benefits of the new algorithm are not of a big magnitude compared to the *number field sieve*, *index calculus* algorithms but it might open a new breach in the stagnant Discrete Logarithm Problem. The introduced Proposition might have some further applications but is not yet clear from the results here.

Acknowledgments

Thanks go to Gunnar Hartung from Karlsruhe Institute of Technology (KIT) for valuable comments on previous versions of this work.