On the Use of the Negation Map in Pollard-Rho Method

CS259C - Elliptic Curves in Cryptography

Final paper
Fall 2011

In order to attack a cryptographic system that does not appear to have any exploitable flaw, one is reduced to solve a discrete logarithm problem, that is, in the context of elliptic curves, given P, Q two points on an elliptic curve, finding $a \in [1, r-1]$ (r being the order of P) such that $Q = aP$. One method to do so is to construct a pseudo-random walk $W_i = a_i P + b_i Q \in \langle P \rangle$ so that finding a is reduced to finding a collision $W_i = W_j$ (this result only holds with high probability: provided $i \neq j$, the probability that a collision $W_i = W_j$ will not result into a value a such that $Q = aP$ is $\mathbb{P}(b_i = b_j)$, non-zero but negligible). The algorithm takes an average of $\sqrt{\frac{\pi r}{2}}$ steps.

One approach to speed-up the process is to use the fact that the negation map $R \mapsto -R$ defines a action on the group $\langle P \rangle$ and would allow, a priori, to divide by 2 the size of the space on which we apply the Pollard-rho algorithm (by defining the random walk on $\langle P \rangle / \{1, -1\}$ instead of $\langle P \rangle$) and so speed-up the process by a factor $\sqrt{2}$. The goal of the present paper is to present this idea in more detail, along with the difficulties that arise in the course of its implementation and the way, proposed by Bos et al in [BKL10] to address them. It follows mainly [BKL10]. As it would be impossible to develop in a couple of pages the entire entire work by Bos et al, the present paper focuses on the Pollard-rho complexity, the implementation of the negation map method, main problem it creates and the main idea underlying the solution proposed by Bos et al. In particular, the previous approaches proposed in the literature to tackle the problem and the subtleties of the solution proposed by Bos et al are beyond the scope of the current paper.

1 Birthday theorem and pseudo-random functions

This section is heavily based on the beginning of the chapter 14 of [Gal]. The proof of the birthday theorem is exactly the same (I simply develop the analysis part, left out in the book).

This section deals with the birthday theorem and how it provides an upper bound for the average complexity for Pollard-rho algorithm. This is not an exact copy of the theorem in [Gal] since its formulation did not seem precise enough*. I reformulate the theorem as follows, to make it as unambiguous as possible:

Theorem 1.1. *If $Y_1, \ldots, Y_k \in \mathcal{S}$ are uniform, independent random variables on $\mathcal{S}$, and $X = \min(l : Y_l \in \{Y_1, \ldots, Y_{l-1}\})$ is the first collision time, then $\mathbb{E}X \leq \sqrt{\frac{n\pi}{2}} + 2$.*

Proof. Let X the random variable corresponding to the number of points sampled at the first collision. Since the points are sampled with uniform probability, the probability for one specific point to be picked is $\frac{1}{n}$. Then, the probability that $l \leq n$ points have been sampled without collision is $\mathbb{P}(X > l) = (1 - \frac{1}{n}) \ldots (1 - \frac{l-1}{n})$ (after k distinct points have been sampled, the $k+1$st point has to be sampled in the $n-k$ remaining points, for $k < l$). Since $x \mapsto \exp(-x)$ is convexe, $\forall x, 1 - x \leq \exp(-x)$ and $\mathbb{P}(X > l) \leq \prod_{i=0}^{l-1} \exp(-i/n) = \exp\left(-\sum_{i=0}^{l-1} i/n\right) = \exp\left(-\frac{l(l-1)}{2n}\right) \leq \exp\left(-\frac{(l-1)^2}{2n}\right)$ (the last inequality coming from the fact that exponential is increasing and $-(l-1) \geq -l$). If $l > n$, $\mathbb{P}(X > l) = 0$

We know that $\mathbb{P}(X > l) = \sum_{i=l+1}^{\infty} \mathbb{P}(X = i)$. Thus,

$$\mathbb{P}(X = l) = \mathbb{P}(X > l - 1) - \mathbb{P}(X > l)$$

and so,

$$\begin{aligned}
\mathbb{E}X &= \sum_{l=1}^{\infty} l\mathbb{P}(X = l) = \sum_{l=1}^{\infty} l\mathbb{P}(X > l-1) - l\mathbb{P}(X > l) = \sum_{l=1}^{\infty} l\mathbb{P}(X > l-1) - \sum_{l=1}^{\infty} l\mathbb{P}(X > l) \\
&= \sum_{l=0}^{\infty} (l+1)\mathbb{P}(X > l) - \sum_{l=1}^{\infty} l\mathbb{P}(X > l) = \mathbb{P}(X > 0) + \sum_{i=1}^{\infty} \mathbb{P}(X > l) \leq 1 + \sum_{l=1}^{n} \exp\left(-\frac{(l-1)^2}{2n}\right) \\
&\leq 2 + \sum_{l=1}^{n-1} \exp\left(-\frac{l^2}{2n}\right)
\end{aligned}$$

Let's now compute an upper bound for the previous sum of exponential: since $x \mapsto \exp(-x)$ is decreasing, we have $\exp\left(-\frac{l^2}{2n}\right) \leq \int_{l-1}^{l} \exp\left(-\frac{x^2}{2n}\right) \mathrm{d}x$ and so, we obtain the inequality: $\sum_{l=1}^{n-1} \exp\left(-\frac{l^2}{2n}\right) \leq \int_0^{n-1} \exp\left(-\frac{x^2}{2n}\right) \mathrm{d}x \leq \int_0^{\infty} \exp\left(-\frac{x^2}{2n}\right) \mathrm{d}x$. But using the

*Indeed, the formulation in [Gal] is : *Let S be a set of N elements. If elements are sampled uniformly at random from S then the expected number of samples to be taken before some element is sampled twice is less than $\sqrt{\pi N/2} = 1.253$.* If X, defined in the proof in [Gal], is understood to be the number of points sampled *before* the point creating the collision is added, then, $\mathbb{P}(X = 1) = \frac{1}{n}$. But following the result and notations in [Gal], $\mathbb{P}(X > 1) = p_{n,1} = 1$ which is wrong. It seems better to define X as I do, as the number of points sampled at collision. Then, $\mathbb{P}(X > 1) = 1$ since, to have a collision, one needs to sample at least two points.

change of variables $u = \frac{x}{\sqrt{n}}$ we obtain $\int_0^\infty \exp\left(-\frac{x^2}{2n}\right) \mathrm{d}x = \sqrt{n}\int_0^\infty \exp\left(-\frac{u^2}{2}\right) \mathrm{d}u$. Now to compute this integral, let's remark that, using polar coordinates, we get:

$$\int_{\mathbb{R}}\int_{\mathbb{R}} \exp\left(-\frac{x^2+y^2}{2}\right) \mathrm{d}x\mathrm{d}y = \int_0^\infty 2\pi r \exp\left(-\frac{r^2}{2}\right) \mathrm{d}u = 2\pi\left[-\exp\left(-\frac{r^2}{2}\right)\right]_0^\infty = 2\pi$$

But, by separation of variables,

$$\int_{\mathbb{R}}\int_{\mathbb{R}} \exp\left(-\frac{x^2+y^2}{2}\right) \mathrm{d}x\mathrm{d}y = \int_{\mathbb{R}} \exp\left(-\frac{x^2}{2}\right) \mathrm{d}x \int_{\mathbb{R}} \exp\left(-\frac{y^2}{2}\right) \mathrm{d}y = \left(2\int_0^\infty \exp\left(-\frac{u^2}{2}\right) \mathrm{d}u\right)^2$$

Finally, $\int_0^\infty \exp\left(-\frac{u^2}{2}\right) \mathrm{d}u = \sqrt{\frac{\pi}{2}}$ and so $\mathbb{E}X \leq 2 + \sqrt{\frac{n\pi}{2}}$ □

Now, in order to get a bound on the average number of steps needed by the Pollard-rho algorithm to find a collision, we need an extra assumption. In [Gol], a pseudorandom function is defined as follows:

> A pseudorandom function is an efficient (deterministic) algorithm that given an n-bit *seed*, s, and an n-bit *argument*, x, returns an n-bit string, denoted $f_s(x)$ so that it is infeasible to distinguish the values of f_s for a uniformly chosen $s \in \{0,1\}^n$, from the values of a truly random function $F : \{0,1\}^n \to \{0,1\}^n$.

Thus, if we were using a pseudorandom function in the definition of the Pollard-rho algorithm, the theorem 1.1 would imply that the average number of steps for a collision to appear is upper bounded by $2 + \sqrt{\frac{n\pi}{2}}$. As we'll see, however, in the definition of the algorithm, a lack of independence at collision time makes this bound slighty too optimistic.

2 Pollard-rho algorithm

The random walk for the Pollard-rho algorithm is developped as follows in [BKL10]: consider a function $\swarrow : \langle P\rangle \to [0, r-1]$ for some r. It induces the partition of $\langle P\rangle$ by $(O_i)_{i\in[0,r-1]} = (\swarrow^{-1}(i))_{i\in[0,r-1]}$. We assume that the $|O_i|$ are close to $\frac{n}{r}$. Then, for each $i \in [0, r-1]$, we compute $u_i, v_i \xleftarrow{\mathrm{R}} [1, n-1]$. We then define the function $f(R) = R + u_{\swarrow(R)}P + v_{\swarrow(R)}Q \in \langle P\rangle$. l is assumed to be a pseudo-random function[†]. Thus, the probability for a certain point of the walk to be in O_i is $p_i = \frac{|O_i|}{n}$.

[†]In [BKL10], it is only assumed that "$\swarrow$ is perfectly random". However, in this context, this seems the only interpretation and it would mean that the image by $\swarrow$ of the successive points in the random walk are independent and identically distributed.

However, as Bos et al point it out, successive iterations of the function f do not yield a pseudo-random walk for the following reason: let's define the class of a point the image by $\swarrow$ of the predecessor of the first occurence of the point. For example, if P_i never appeared in the walk, its class would be $\swarrow(P_{i-1})$; if it appeared first as $P_j, j < i$ (that is, j is the smallest integer such that $P_i = P_j$), then the class of P_i is $\swarrow(P_{j-1})$. Since the function $\swarrow$ is pseudo-random, $\mathbb{P}(\swarrow(R) = i) = p_i$ for $R \in \langle P \rangle$. But, as Bos et al. notice, if the first collision point is of class i, the preceding point is not in O_i. Indeed, if $P_{k+1} = P_{l+1}$ is the first collision (P_{k+1} being the first occurence of P_{l+1}) and is of class i, then, $\swarrow(P_k) = i$. If $\swarrow(P_l) = i$, then $P_k = P_{k+1} - u_i P - v_i Q = P_{l+1} - u_i P - v_i Q = P_l$ and so, $P_{k+1} = P_{l+1}$ is not the first collision. Thus, if there is a collision at step $l+1$, then, the class of P_{l+1} and $\swarrow(P_l)$ have to be different. But, "P_{l+1} has class i" and $\swarrow(P_l) = i$ are two independent events at collision time $l+1$ (since they depend respectively on points P_k and P_l with $k < l$ (since $P_{k+1} \neq P_{l+1}$ is the first appearance of P_{l+1}) and so $\swarrow(P_l)$ and $\swarrow(P_k)$ are independent since $\swarrow$ is pseudo-random). Thus, the joint probability is

$$\mathbb{P}(\text{class}(P_{l+1}) = i \& \swarrow(P_l) = i) = \mathbb{P}(\text{class}(P_{l+1}) = i)\mathbb{P}(\swarrow(P_l) = i) = p_i^2$$

Thus, the probability of a collision at $l+1$ is $\frac{l}{n}\left(1 - \sum_i p_i^2\right)$. We can account for this imperfection in the independence of the distribution of the points on the random walk in the proof of the birthday theorem by replacing the factor n by $\frac{n}{\left(1-\sum_i p_i^2\right)}$ in the computation of $\mathbb{P}(X > l)$ (since the only thing we use to compute it the probability of a collision at time $l+1$), obtaining the bound on the average number of steps:

$$2 + \sqrt{\frac{n\pi}{2\left(1 - \sum_i p_i^2\right)}}$$

If the random walk start on a point of form $a_0 P + b_0 Q$, then, for all i, $P_i = a_i P + b_i Q$ (with the recursion formula $a_{i+1} = a_i + u_{\swarrow(P_i)}$ and $b_{i+1} = b_i + v_{\swarrow(P_i)}$)) and so, given a collision $P_i = P_j$, provided $b_i \neq b_j$ (which happens with high probability), we obtain: $Q = \frac{a_j - a_i}{b_i - b_j} P$, which solve the discrete log problem.

3 Negation map

This part is mainly based on the exercise 7 of the homework 3. It is a particular case of the ideas developped in [WZ98].

This idea underlying the use of negation map is the following: we start by considering the action of $H = \{1, -1\}$ on $\langle P \rangle$. Now, given the function $f : \langle P \rangle \to \langle P \rangle$, we can construct the equivariant function $\hat{f} : R = (x, y) \mapsto f((x, \min(y, -y)))$(where the min

function is applied to the representatives of y and $-y$ in $[0, p-1]$)[‡]. Then, a random walk in $\langle P \rangle$ yields naturally a random walk in $\langle P \rangle / H$. Then, it is possible to apply the Pollard-rho algorithm to the walk on $\langle P \rangle / H$ which has half the size of $\langle P \rangle$ and which, theoretically, should gain a factor $\sqrt{2}$ in finding a collision. Then, from a collision in $\langle P \rangle / H$, it is easy to recover a collision in $\langle P \rangle$: indeed, if $(\hat{P}_i)$ is the random walk on $\langle P \rangle / H$ and $\hat{P}_i = \hat{P}_j$, then, $P_i = \pm P_j$ and using the notations in the previous section, we obtain (again, with high probability):

$$Q = \frac{a_j \mp a_i}{b_i \mp b_j} P$$

Thus, this would yield the wanted result with a gain factor of $\sqrt{2}$.

Unfortunately, this random walk on $\langle P \rangle / H$ also introduces small cycles with a non-negligible probability, preventing the algorithm to converge. The goal of the rest of this paper is to present the how such cycles may appear along with main idea underlying the approach Bos et al followed to address the problem without impacting too much the performance of the algorithm.

4 Fruitless cycles

This section is, in part, based on the exercise 7 of homework 3. In this section, I assume that $P_i = a_i P + b_i Q$ (and I keep track of the a_i, b_i)

A fruitless cycle is a cycle leading to a collision $P_k = P_l$ such that $b_k = b_l$ (or a collision $P_k = -P_l$ such that $b_k = -b_l$). In particular, it carries no information about the discrete log.

Let's consider the simplest fruitless cycle: the 2-cycle. Let's assume that $P_l = (x_l, y_l)$ $y_l < -y_l$ and $y_{l+1} > -y_{l+1}$ (which happens with probability about $\frac{1}{4}$). Let's also assume that $\mathcal{A}(P_l) = \mathcal{A}(P_{l+1}) = i$ for some $i \in [0, r-1]$ (that happens with probability about $\frac{1}{r}$ since we assumed the $|O_i|$ roughly of equal size). Then, $P_{l+2} = -P_{l+1} + u_i P + v_i Q = -(P_l + u_i P + v_i Q) + u_i P + v_i Q = -P_l$. Thus, for the induced random walk in $\langle P \rangle / H$, it implies that $\hat{P}_{l+2} = \hat{P}_l$ (and since $b_{l+2} = -b_{l+1} + u_i = -(b_l + u_i) + u_i = -b_l$ this cycle is fruitless). This cycle appear with probability at least $\frac{1}{4r}$ and since the result would be the

[‡]This definition, which can be applied to any function f is not exactly the one used in [BKL10]: indeed, if $f(R) = R + S$, as it is defined in [BKL10] for the regular Pollard-rho algorithm, and if $R = (x, y)$ and $y > -y$, then, in our definition, $\hat{f}(R) = f(-R) = -R + S$ while in their definition, the image of R after one step would be $-R - S$. Although it may change slightly the order of hypotheses resulting in cycles, this does not change anything to the conclusions. Also, strictly speaking, the previous construction only gives us a map $\hat{f} : \langle P \rangle / H \to \langle P \rangle$. However, it is easy to make it a map $\langle P \rangle / H \to \langle P \rangle / H$ by post-composing by the quotient map (which, in practice we don't have to do).

same in the reverse situation ($y_i > -y_i$ and $y_{i+1} > -y_{i+1}$), a fruitless 2-cycle appears with probability at least $\frac{1}{2r}$. Since the algorithm performs optimally for $\log(r) \leq 10$ (based on Fig. 1 in [BKL10]), r cannot be to big and thus, this probability is not negligible.

Using a similar approach, it is possible to see that 4-cycles may also appear (though with a smaller probability). To show that, let's consider the following notation (mainly from Bos et al) to make the proof simpler: We attach to each point $R(x, y)$ the signature $+$ or $-$ depending on whether $y < -y$ or $y > -y$ and the number $i = \mathscr{l}(R)$. We also denote $S_i = u_i P + v_i Q$. Thus, if a point R has a signature $(-, i)$, then $\hat{f}(R) = -R + S_i$: we denote it $R \xrightarrow{(-,i)} -R + S_i$. For example, the previous case can be rewritten as follows:

$$R \xrightarrow{(+,i)} R + S_i \xrightarrow{(-,i)} -R$$

or, inverting the signs:

$$R \xrightarrow{(-,i)} -R + S_i \xrightarrow{(-,i)} R$$

yielding the 2-cycle in $\langle P \rangle / H$[§]. Then, using this notation, we can see that a 4-cycle may appear as follows:

$$\begin{aligned}
R &\xrightarrow{(+,i)} R + S_i \xrightarrow{(-,j)} -R - S_i + S_j \xrightarrow{(+,i)} -R + S_j \xrightarrow{(-,j)} R \\
R &\xrightarrow{(-,i)} -R + S_i \xrightarrow{(-,j)} R - S_i + S_j \xrightarrow{(+,i)} R + S_j \xrightarrow{(-,j)} -R \\
R &\xrightarrow{(+,i)} R + S_i \xrightarrow{(+,j)} R + S_i + S_j \xrightarrow{(-,i)} -R - S_j \xrightarrow{(+,j)} -R \\
R &\xrightarrow{(-,i)} -R + S_i \xrightarrow{(+,j)} -R + S_i + S_j \xrightarrow{(-,i)} R - S_j \xrightarrow{(+,j)} R
\end{aligned}$$

i can be any element in $[0, r-1]$ and we need $j \neq i$ (hence the factor $\frac{r-1}{2r}$ resulting from the assumption for the second step). Thus the probability for this 4-cycle to appear is at least $\frac{r-1}{4r^3}$.

It can be shown in the same way that more complicated cycle may appear, making the loop detection rather delicate. The next section presents the heuristic at the center of [BKL10] to avoid getting trapped in a cycle.

[§]It is not exactly the same sequence as in [BKL10] since I chose for $\hat{f}$ the convention defined in the homework (making $\hat{f}$ equivariant), which is slightly different from the definition of $\hat{f}$ in [BKL10]. However, the principle is strictly the same

5 Fruitless cycles handling

The following heuristic is from [BKL10], section 3.3.

Heuristic 5.1. *A cycle with at least one doubling is most likely not fruitless i.e., a collision resulting from one such cycle is likely to carry the solution of the discrete log problem.*

Proof. The heuristic argument given by Bos et al is little convincing. They consider starting from a point $uP + vQ$ in the cycle and apply a certain number of times the walk along with some doubling and negations. Thus, if c doubling are applied along with regular steps, then, when the walk cycles back to $uP + vQ$, we obtain the following equality:

$$uP + vQ = \pm 2^c(uP + vQ) + \sum_{i=0}^{r-1} c_i u_i P + c_i v_i Q$$

The left hand side stands for the original point and the right hand-side stands for a general expression of the point of the walk after many steps and c doubling (since, by distributivity, doubling a point in the walk is doubling the initial point and doubling the sum of everything that was added in the meantime). This leads to the expression:

$$\left((1 \mp 2^c)u - \sum_{i=0}^{r-1} c_i u_i\right) P + \left((1 \mp 2^c)v - \sum_{i=0}^{r-1} c_i v_i\right) Q = 0$$

The cycle would be fruitless if we could not deduce the discrete logarithm from the collision, that is if $\left((1 \mp 2^c)v - \sum_{i=0}^{r-1} c_i v_i\right) = 0$ (which is this case is equivalent to $\left((1 \mp 2^c)u - \sum_{i=0}^{r-1} c_i u_i\right) \neq 0$). What is slightly puzzling is the fact that they can infer from $1 \mp 2^c \neq 0$ (which would be equally true if $c = 0$) that $\left((1 \mp 2^c)u - \sum_{i=0}^{r-1} c_i u_i\right)$ "is most likely not divisible by the group order".

However, this heuristic argument points out a very interesting fact: doubling has an multiplicative impact on the quantity $\left((1 \mp 2^c)v - \sum_{i=0}^{r-1} c_i v_i\right)$ while each step has an additive impact on it. Thus, it seems reasonable to assume that we can prevent it to be zero by multiplications by 2. Furthermore, as the results in [BKL10] suggest, the method of doubling performs better than other methods for handling fruitless cycles.

□

In spite of some puzzling aspect in the heuristic argument given by Bos et al, this presents us with an interesting and empirically promising method to deal with cycles using negation map in Pollard-rho algorithm.

6 Limitation and future work

The main limitation of this result is that, as it is heuristically relying on doubling, it is harder to implement in an SIMD model (as it would introduce branching in the code unless all the walks are doubled at the same time, which may not have the expected effect), which limits its applicability in real situations. This problem is addressed in a paper written by Bernstein et al ([BLS11]).

References

[BKL10] Joppe Bos, Thorsten Kleinjung, and Arjen Lenstra. On the use of the negation map in the pollard rho method. In Guillaume Hanrot, François Morain, and Emmanuel Thom, editors, *Algorithmic Number Theory*, volume 6197 of *Lecture Notes in Computer Science*, pages 66–82. Springer Berlin / Heidelberg, 2010.

[BLS11] Daniel Bernstein, Tanja Lange, and Peter Schwabe. On the correct use of the negation map in the pollard rho method. In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *Public Key Cryptography PKC 2011*, volume 6571 of *Lecture Notes in Computer Science*, pages 128–146. Springer Berlin / Heidelberg, 2011.

[Gal] Steven Galbraith. *Mathematics of Public Key Cryptography*. Online version 1.1.

[Gol] Oded Goldreich. *Foundations of Cryptography (a primer)*. Revision: Feb. 2005.

[WZ98] Michael J. Wiener and Robert J. Zuccherato. Faster attacks on elliptic curve cryptosystems. In *Selected Areas in Cryptography, LNCS 1556*, pages 190–200. Springer-Verlag, 1998.