On Splitting a Point with Summation Polynomials in Binary Elliptic Curves

Nicolas T. Courtois

University College London, Computer Science, Room 6.18. Gower Street, WC1E 6BT, London, UK
n.courtois@cs.ucl.ac.uk

Abstract. Recent research for efficient algorithms for solving the discrete logarithm (DL) problem on elliptic curves depends on the difficult question of the feasibility of index calculus which would consist of splitting EC points into sums of points lying in a certain subspace. A natural algebraic approach towards this goal is through solving systems of non linear multivariate equations derived from the so called summation polynomials which method have been proposed by Semaev in 2004 [12].
In this paper we consider simplified variants of this problem with splitting in two or three parts in binary curves. We propose three algorithms with running time of the order of $2^{n/3}$ for both problems. It is not clear how to interpret these results but they do in some sense violate the generic group model for these curves.

Key Words: cryptanalysis, summation polynomials, algebraic attacks, block ciphers, Gröbner bases, DL problem, finite fields, elliptic curves, ECDSA, generic group model.

1 Motivation: Solving Semaev Systems of Equations

Efficient algorithms for solving so called Semaev equations, which can be defined informally as systems of equations built on the basis of the summation polynomials over elliptic curves [12, 13] is a basic tool for cryptanalysis and a high profile open problem in modern cryptanalysis. Even partial results are valuable and could be exploited inside or extended into a full attack on the DL problem on elliptic curves.

1.1 Point Splitting

Let $R = (R_X, R_Y)$ be the target point on an elliptic curve which we want to split in a form

$$R = P_1 + \ldots P_i + \ldots + P_t$$

with some t points P_i, $i = 1..t$, all lying on the same elliptic curve with (preferably) all the P_i lying in some well-chose subspace.

This problem has attracted considerable attention in the last 10+ years, and researchers have tried to encode this problem as a problem of solving a certain system of equations known as summation polynomials, sometimes also called Semaev polynomials, cf. [12, 11, 5, 8].

More recently in 2015 a particular way to re-write this problem using only the simplest non-trivial summation polynomial S_3 and with many additional variables have been proposed by Semaev in 2015, cf. [13]. We recall this particular way to encode the problem of splitting the point on elliptic curve. We call x_i the x coordinate of point P_i in $GF(2^n)$ and let u_i be $t-2$ auxiliary variables in $GF(2^n)$.

$$\begin{cases} S_3(u_1, x_1, x_2) \\ S_3(u_1, u_2, x_3) \\ \vdots \\ S_3(u_i, u_{i+1}, x_{i+2}) \\ \vdots \\ S_3(u_{t-3}, u_{t-2}, x_{t-1}) \\ S_3(u_{t-2}, x_t, R_X). \end{cases}$$

We have $t-1$ equations in $GF(2^n)$ where t is the number of points in our decomposition of R as a sum t elliptic curve points. We call it Semaev-serial system of equations as it effectively is a serial connection[1] of several systems of equations of type S_3 in a certain encoding (with a topology of straight line with connections only between consecutive components).

[1] This sort of topology for systems of equations is very common in block cipher cryptanalysis [1–3] and the hardness of solving these systems can be seen as a hardness to derive ANY sort of algebraic or statistical knowledge about the middle variables, not easily accessible for the attacker.

1.2 Basic Point Splitting Problem

If a complex problem is efficiently solvable[2], the easier one should also be solvable. The most basic problem which we would like to be able to solve using the Semaev polynomials is the problem of splitting a point in two:

$$S_3(x_1, x_2, R_x)$$

where P_1, P_2 would lie in a well-chosen subspace and R is the target point.

In this paper we focus on binary elliptic curves. We impose no other restriction. We will also assume than n is odd which is however a conservative choice (no no-trivial sub-field) and which is also believed to be the hard case, much more likely to be recommended in any sort of application (prime n would typically be preferred).

As in other works on this topic [12, 13] our preferred choice for the subspace is $G_n^{1/2}$ which we define as follows.

Definition 1 **($G_n^{1/2}$).**
Let G_n be a vector space which contains all polynomials in $GF(2)[T]$ with degree $< n$ and coefficients in $GF(2)$.
Let $G_n^{1/2}$ be a sub-space of all univariate binary polynomials with degree $< n/2$. Furthermore, let $P(T)$ is the irreducible polynomial used to define $GF(2^n)$. We identify $G_n^{1/2}$ with a sub-space of $GF(2^n)$ and we say that we have $x \in G_n^{1/2} \subset GF(2^n)$ if its polynomial representation modulo $P(T)$ has degree $< n/2$

The main goal of this paper is to solve this single point splitting problem with 2 points in $G_n^{1/2}$ faster than in $2^{n/2}$ time.

1.3 Point Multiple Basic Splitting Problem

In this paper we also study a very closely related variant of our problem in which we are allowed to multiply the target point by a [known] scalar as follows.

Definition 2 (Generalized Point Splitting Problem). *Let R be a target point on the elliptic curve. Find a [preferably quite small] integer $K \in \mathbb{Z}$ and $X, Y \in B_n^{1/2} \subset GF(2^n)$ such that*

$$S_3(X, Y, S)$$

with S being a suitable scalar multiple of the target point R with $S = R.K$.

[2] A recent survey paper [6] from October 2015 says that "there is no consensus" whether this is the case.

2 Point Splitting in Two in Binary Curves

2.1 Summation Polynomials in Characteristic 2

Following [12, 13] In the case of binary elliptic curves $E(GF(2^n))$ the S_3 equation is the following equation with 3 variables over $GF(2^n)$:

$$S_3(x_1, x_2, x_3) = (x_1x_2 + x_1x_3 + x_2x_3)^2 + x_1x_2x_3 + B$$

where B is the coefficient of the elliptic curve in question in a popular notation, cf. [7, 12, 13].

2.2 Point Multiple Splitting in Two in Binary Curves

We recall that addition and squaring commute in $GF(2^n)$ and that additions are modulo 2. Therefore we have the following equation to solve with 2 variables in $G_n^{1/2} \subset GF(2^n)$.

$$\begin{cases} X^2Y^2 + S_X \cdot XY = S_X^2(X+Y)^2 + B \\ S = K.R \end{cases}$$

2.3 Trace Function(s)

We view $GF(2^n)$ as a Galois extension of $GF(2)$, again let n be odd. We define, for any constant F in $GF(2^n)$ or any polynomial F in $GF(2^n)$[some vars] the standard trace operator which we will split in two halves. Let $n = 2k + 1$.

$$\begin{cases} Tr(F) = \sum_{\sigma \in G} \sigma F & G = \text{Galois group of } GF(2^n) \\ Tr(F) = \sum_{i=0}^{n-1} F^{2^i} & n \text{ terms} \\ Tr_0(F) = \sum_{i=0}^{k-1} F^{2^{2i}} + F^{2^{2k}} & k+1 \text{ even terms} \\ Tr_1(F) = \sum_{i=0}^{k-1} F^{2^{2i+1}} & k \text{ odd terms} \\ Tr(F) = Tr_0(F) + Tr_1(F) & \\ Tr_1(F^2) = F + Tr_0(F) & \\ Tr_i(F^2) = (Tr_i(F))^2 & \text{commutes, both } i = 0 \text{ and } 1 \\ Tr_1(F + F^2) = F + Tr(F) & \end{cases}$$

2.4 Re-writing Our S3 Problem

Our equation is:

$$X^2Y^2 + S_X \cdot XY = S_X^2(X+Y)^2 + B.$$

We assume that $S_X \neq 0$ which is almost always true, and we rewrite our equation to be solved:

$$X^2Y^2/S_X^2 + XY/S_X = (X+Y)^2 + B/S_X^2$$

with 2 variables in $G_n^{1/2} \subset GF(2^n)$ and a modified target S_x. We apply Tr_1 to both sides and use the fact that $Tr_1(F+F^2) = F + Tr(F)$ with $F = XY/S_X$. Furthermore $Tr(F) \in \{0,1\}$ which part is very easy to guess for the attacker and which we denote by [+1], an optional addition of 1, to be applied [or not].

$$XY/S_X[+1] = Tr_1((X+Y)^2 + B/S_X^2)$$

We multiply by S_X:

$$XY[+S_X] = S_X \cdot Tr_1((X+Y)^2 + B/S_X^2)$$

Overall we want to solve:

$$\begin{cases} XY[+S_X] = S_X \cdot Tr_1^2(X+Y+\sqrt{B}/S_X) \\ S = K.R \end{cases}$$

with K being a certain integer and $X, Y \in G_n^{1/2} \subset GF(2^n)$.

It is easy to see that for any K this problem contains n bits of information about the solution and there is $n/2 + n/2$ variables. We expect that on average it has 1 solution[3] X, Y.

2.5 Our First Attack - Splitting in Two

Our attack is as follows.

1. We try many different multiples $S = K.R$ for $K = 1 \ldots 2^{n/6}$.
2. Each S has on average 1 solution X, Y for the strict point splitting problem. However we are going only to look at solutions with additional properties.
3. We represent X, Y as univariate polynomials of degree up to $n/2$ in $GF(2)[T]$. We consider the polynomial $GCD(X, Y)$ and we are interested in the probability that polynomials X and Y in $GF(2)[T]$ have a common divisor M of degree exactly[4] $\lfloor n/6 \rfloor - 1$ in $GF(2)[T]$.
   It is easy to see that this probability is roughly[5] about $2^{-n/6}$.
4. Our goal is to solve the equation $S_3(X, Y, S_X)$ in this special case with a shared divisor, which will show to be easier than the general point splitting. We expect that on average there will be about one multiple $S = K.R$ for which this works and such M exists.
5. To summarize, the attacker tries many different $K = 0 \ldots 2^{n/6}$ and expects that $M|X$ and $M|Y$ for some K.
6. For each guess for K, we also need to guess the value of M which is also correct with probability $2^{-n/6}$.

[3] To simplify our analysis, we ignore the fact that solutions typically come in pairs as the problem is symmetric, so sometimes it has 0 solutions, sometimes 2, and less frequently it has 1 or a different small number of solutions.

[4] We leave for future research to see that this assumption can be relaxed.

[5] In fact it is smaller by a small constant factor. In this paper we only do a simplified asymptotic complexity analysis, this point requires a more detailed analysis.

7. Overall both guesses are correct with probability $2^{-2n/6}$ and all our assumptions hold.
8. For each guess of K, M we proceed as follows.
   (a) We compute the polynomial M^2 which has approximately $2n/6-1$ bits.
   (b) If our guess is correct $M^2|XY$. Let $X = MX'$ and $Y = MY'$. Here $X', Y' \in G_n^{2/6}$ and $M \in G_n^{1/6}$.
   (c) We have $XY[+S_X] = S_X \cdot Tr_1^2(X + Y + \sqrt{B}/S_X)$ Therefore
   $$M^2X'Y'[+S_X] = S_X \cdot Tr_1^2(M(X' + Y') + \sqrt{B}/S_X)$$
   (d) Let $L \in GF(2^n)$ be the left hand side of the equation above. We recall that if our assumptions are correct, no modular reduction modulo $P(T)$ occurs inside the product $M^2X'Y'$. Here $X'Y' \in G_n^{4/6}$ and $M^2 \in G_n^{2/6}$.
   (e) Therefore, given that M^2 is known, we can write $2n/6$ equations on the L_i variables which will be linear or affine, depending whether or not we decided to add the constants of $[+S_X]$. This is a choice made by the attacker which will be valid with probability $1/2$.
   (f) Let $R \in GF(2^n)$ be the right hand side of the equation above. If our assumptions are correct, we have
   $$R = S_X \cdot Tr_1^2(R' + \sqrt{B}/S_X$$
   with $R' = M(X' + Y')$. Again, no modular reduction occurs inside $R' = M(X' + Y')$. Therefore given the known M value (from our assumption) we can write $n/6$ linear equations on the R'_i variables.
   (g) In addition we have $n/2-1$ linear equations on the R'_i which come from the fact that the degree of $M(X' + Y')$ is at most $n/2 + 1$.
   (h) We observe that the expression $R = S_X \cdot Tr_1^2(R' + \sqrt{B}/S_X$ is a linear bijection for which the attacker knows all the coefficients. Therefore our $n/2 + n/6$ linear equations on the R'_i translate into $n/2 + n/6$ linear equations on the $R_i = L_i$.
   Previously we have already constructed $2n/6$ linear/affine equations on the L_i. These two sets of equations for each side are expected to be distinct and overall we obtain about $n/2 + n/6 + 2n/6 = n$ linearly independent equations on the L_i.
   (i) With n equations and n variables, we recover L and $R = L$.
   (j) Now we need to verify of the solution is consistent: We compute $U = L/M^2$ and we compute R' and we put $V = R'/M$. Now the questions is whether it is possible that $X'Y' = U$ and $X' + Y' = V$. In order to see that we solve the quadratic equation $X'(V - X') = U$ in $GF(2^n)$.
   (k) We check if degrees of X' and $V - X'$ are at most $2n/6$.
9. If the degrees are within bounds, we have found a solution X, Y.
10. Otherwise we re-start and try the same steps again for another case K, M.

Overall our algorithm is expected to check $2^{2n/6}$ cases K, M and one such assumption is correct with probability $2^{-2n/6}$. One is expected to be correct on average. Each case is checked in polynomial time with very small memory.

3 Point Splitting in Three in Binary Curves

After this paper was published on eprint it occurred to us that point splitting in three is even easier. The goal is as follows. Given an arbitrary point R on a binary elliptic curve and find a decomposition such that

$$\begin{cases} R = O + P + Q & \text{in the elliptic curve } E(GF(2^n)) \\ deg(O_X) < n/3 \\ deg(P_X) < n/3 \\ deg(Q_X) < n/3 \end{cases}$$

Here is one simple method to achieve this also in time $2^{n/3} \cdot poly$.

First, it is easy to see that there will be very roughly 1 solution to our problem on average[6]. Then we proceed as follows.

1. We guess O_X with $deg(O_X) = \lfloor n/3 \rfloor - 1$.
2. We compute $T = R - O$ on the elliptic curve.
3. Let T_X be its X coordinate.
4. As before, we need to solve
$$\left\{ XY[+T_X] = T_X \cdot Tr_1^2(X + Y + \sqrt{B}/T_X) \right.$$
but now we have stricter constraints on X, Y which make it a lot easier.
5. Again let L be the left hand side and R the right hand side, we need to find X, Y with such that $L = R$ and $deg(X) < n/3$ and $deg(Y) < n/3$.
$$R = T_X \cdot Tr_1^2(R' + \sqrt{B}/T_X$$
with $R' = X + Y$.
6. We have $deg(XY) < 2n/3$ therefore we get $n/3$ linear/affine equations on L depending whether we add T_X or not [attacker's guess on 1 extra bit].
7. We have $deg(X + Y) < n/3 + 1$ therefore we get $2n/3 - 1$ linear equations on R. Due to the complex affine transformation of type $T_X \cdot Tr_1^2()$ etc, we expect that these equations are disjoint from these coming from L.
8. Overall we get about $n - 1$ linearly independent equations on $L = R$.
9. We obtain 2 solutions for L and we need to verify if any of these solution is consistent.
10. We compute $U = L$ and we compute $V = R'$. Now we solve the quadratic equation $X(V - X) = U$ in $GF(2^n)$ and verify if we have both degrees of X and $V - X$ below $n/3$.
11. This is expected to happen for about one initial choice of O_X.

Overall this algorithm is expected to check $2^{n/3}$ cases O_X and find one solution to our problem. Each case is checked in polynomial time with very small memory.

[6] This is in great simplification, in fact the problem is again symmetric and for each solution we have up to 5 more derived solutions obtained by permuting the 3 points, and sometimes there are 0 solutions

4 Point Multiple Splitting with Two Smaller Spaces

Another result we can easily obtain is as follows:

Fact 4.0.1. Given an arbitrary point R on a binary elliptic curve we can find in time $2^{n/3} \cdot poly(n)$ a [relatively] small multiple of this point $K \leq 2^{n/3}$ such that

$$\begin{cases} K.R = P + Q \text{ in the elliptic curve } E(GF(2^n)) \\ deg(P_X) < n/3 \\ deg(Q_X) < n/3 \end{cases}$$

Justification: This can be seen as a variant or synthesis of two previous results. One method to obtain this is to replace in Section 2.5 a search for K, M where both K and M take $2^{n/6}$ values by a search with K taking $2^{n/3}$ values and $M = 1$. No other changes are necessary in Section 2.5.

Another method to obtain the same result is to replace the first step in Section 3 where the attacker is trying many polynomials of low degree $deg(O_X) < n/3$ by $O_X = K.R$ by for all $K \leq 2^{n/3}$ which values O_X will no longer required to be of low degree.

5 Conclusion

In this paper we have introduced a new algorithm which takes as input an arbitrary point R on a binary elliptic curve and finds a relatively small multiple of this point $K \leq 2^{n/6}$ such that

$$\begin{cases} K.R = P + Q & \text{in the elliptic curve } E(GF(2^n)) \\ deg(P_X) < n/2 \\ deg(Q_X) < n/2 \\ P_X \text{ and } Q_X \text{ share a factor of degree } n/6. \end{cases}$$

This problem was motivated by a more general problem of point splitting suggested by Semaev in 2004 [12] with an idea of constructing potentially and in the best scenario[7]] an index calculus algorithm for elliptic curves. In this paper we only consider splitting in two and three and show first non-trivial attacks on this problem. Our algorithms run in time $2^{n/3} \cdot poly(n)$ and require negligible memory. Until now the best known algorithm for splitting in two would be brute force algorithm with running time of $2^{n/2} \cdot poly(n)$.

Similarly, splitting a point in three is even easier and we present an algorithm in time $2^{n/3} \cdot poly(n)$ and small memory which does the following decomposition:

$$\begin{cases} R = O + P + Q & \text{in the elliptic curve } E(GF(2^n)) \\ deg(O_X) < n/3 \\ deg(P_X) < n/3 \\ deg(Q_X) < n/3 \end{cases}$$

We also provide a method for splitting a multiple in **two** with degrees $< n/3$.

5.1 Binary Curves vs. Ideal Groups

A result with only 2 or 3 points may seem weak however we already obtain some sort of violation of the ideal group model for binary curves. Very clearly we are able to use the binary representation of the EC point for a non-trivial algorithmic purpose with running time faster than in generic groups. This could imply that certain cryptographic protocols using binary elliptic curves with 256 bits field size should no longer be claimed to be provably secure in a meaningful sense based on the generic group model. It also means that we should be more conservative and maybe only claim a security level of about 2^{85} for all cryptographic schemes using binary curves. At this moment there are serious questions about how the results of this paper should be interpreted. Currently it is a bit like breaking a block cipher with fewer rounds than the real thing and inevitably not everybody will agree that the current result should be called "an attack" of any sort or is violating a certain precise claim or assumption previously made in crypto literature. Things will become more clear with time as complexity of splitting EC points in 4,5 and more points is studied and we will have a better idea how these results impact the DL problem and security of practical cryptosystems[8].

[7] This is if splitting with sufficiently small spaces can be achieved.

[8] Many of which are not proven secure w.r.t. the DL problem.

5.2 Related Works and Future Research

The existence of yet faster truly sub-exponential algorithms for binary curves have also been conjectured [12, 13, 11, 5] however such a result has not yet been achieved, cf. [4, 6, 8–10, 14]. Recent research in this space can be summarized as follows. It seems that it is totally incorrect to believe that systems of equations such as in Section 1.1 after a Weil descent conversion to a pure GF(2) system of multivariate equations, would have a regularity degree which is constant and does not depend on n. However this does not exclude that such systems of equations could be solved efficiently by other methods, and moreover if the degree of regularity grows with n we could still have sub-exponential complexity. At this moment following [6] “there is no consensus” whether any of these techniques can be made to work in sub-exponential time. In this paper our ambition was more modest. Our current result remains exponential and suggests that if an index calculus algorithm for binary curves can at all be constructed following [12] it could rather have exponential complexity [but lower than with current attacks].

Future Research: It is easy to see that the two simple algorithms we present here can be used again to re-split points obtained from splitting, and we expect that there exists interesting extensions and generalizations of our method for splitting in 4 points and more. At this moment we have not yet achieved a stage where a full index calculus algorithm with linear algebra on spaces we obtain would be at all feasible. Our conjecture is however that there exists an algorithm for solving the full DL problem on binary elliptic curves with running time of order of $2^{n/3}$.

Acknowledgments:

This research was inspired by numerous technical discussions on the security of ECCs I had in the last 2 years with Christophe Petit and Jean-Jacques Quisquater. I would like to thank Jonathan Bootle, Steven Galbraith and Louis Goubin for extremely helpful feedback and comments.

References

1. Nicolas Courtois, Gregory V. Bard: *Algebraic Cryptanalysis of the Data Encryption Standard,* In Cryptography and Coding, 11-th IMA Conference, pp. 152-169, LNCS 4887, Springer, 2007.
2. Nicolas T. Courtois: *How Fast can be Algebraic Attacks on Block Ciphers?* In online proceedings of Dagstuhl Seminar 07021, *Symmetric Cryptography* 7-12 January 2007, also available from `http://eprint.iacr.org/2006/168/`.
3. Nicolas Courtois: *Algebraic Complexity Reduction and Cryptanalysis of GOST*, preprint, 2010-2015, available at `http://eprint.iacr.org/2011/626`.
4. Claus Diem: *On the discrete logarithm problem in elliptic curves,* In Compos. Math., 147(2011), pp. 75-104.
5. Jean-Charles Faugère, Ludovic Perret, Christophe Petit, and Gwenaël Renault: *Improving the complexity of index calculus algorithms in elliptic curves over binary fields,* In Eurocrypt 2012, LNCS 7237, pp. 27-44, Springer 2012.
6. Steven D. Galbraith, Pierrick Gaudry: *Recent progress on the elliptic curve discrete logarithm problem,* preprint, 22 Oct 2015, `https://eprint.iacr.org/2015/1022.pdf`
7. Darrel Hankerson, Alfred Menezes, Scott Vanstone: *Guide to Elliptic Curve Cryptography,* book, hardcover, 331 pages, Springer 2004.
8. T. Hodges, C. Petit, and J. Schlather: *First fall degree and Weil descent,* In Finite Fields Appl., 30(2014), pp. 155-177.
9. Michiel Kosters, Sze Ling Yeo: *Notes on summation polynomials,* revised 8 Jun 2015 `http://arxiv.org/abs/1503.08001`
10. Ming-Deh A. Huang, Michiel Kosters, Sze Ling Yeo: *Last Fall Degree, HFE, and Weil Descent Attacks on ECDLP,* In Crypto 2015, LNCS 9215, pp. 581-600, August 2015.
11. Christophe Petit and Jean-Jacques Quisquater: *On polynomial systems arising from a Weil descent,* In Asiacrypt 2012, LNCS 7658, pp. 451-466, Springer 2012.
12. Igor Semaev: *Summation polynomials and the discrete logarithm problem on elliptic curves,* Preprint, available at `eprint.iacr.org/2004/031/`.
13. Igor Semaev: *New algorithm for the discrete logarithm problem on elliptic curves,* Preprint, available at `eprint.iacr.org/2015/310/`.
14. M. Shantz and E. Teske: *Solving the elliptic curve discrete logarithm problem using Semaev polynomials, Weil descent and Gröobner basis methods - an experimental study,* In LNCS 8260, pp. 94-107, Springer 2013.