Index calculus method for solving elliptic curve discrete logarithm problem using quantum annealing

Michał Wroński[0000−0002−8679−9399]

Military University of Technology, Kaliskiego Str. 2, Warsaw, Poland
michal.wronski@wat.edu.pl

Abstract. This paper presents an index calculus method for elliptic curves over prime fields using quantum annealing. The relation searching step is transformed into the QUBO (Quadratic Unconstrained Boolean Optimization) problem, which may be efficiently solved using quantum annealing, for example, on a D-Wave computer. Unfortunately, it is hard to estimate the complexity of solving the given QUBO problem using quantum annealing. Using Leap hybrid sampler on the D-Wave Leap cloud, we could break ECDLP for the 8-bit prime field. The most powerful general-purpose quantum computers nowadays would break ECDLP at most for a 6-bit prime using Shor's algorithm. In presented approach, the Semaev method of construction of decomposition base is used, where the decomposition base has a form $\mathcal{B} = \left\{x : 0 \leq x \leq p^{\frac{1}{m}}\right\}$, with m being a fixed integer.

Keywords: Cryptanalysis · index calculus on elliptic curves · quantum annealing · QUBO · D-Wave.

1 Introduction

Quantum computing is a significant branch of modern cryptology. As was presented in the introduction, several quantum algorithms support the cryptanalysis of public-key schemes. In the last few years, notable progress in this field has been established. The two approaches of quantum computing for cryptography are the most popular nowadays. The first one is an approach using quantum annealing, which is used in D-Wave computers. The second one is the general-purpose quantum computing approach. The first approach has limited applications, where mainly QUBO and Ising problems may be solved using such quantum computers. On the other hand, several cryptographical problems may be translated to the QUBO problem, for example, integer factorization. The quantum factorization record belonged to the D-Wave computer for some time, where Dridi and Alghassi [3] factorized integer $200,099$. This result was later beaten by Jiang et al. [6], where $376,289$ was factorized, and by Wang et al. [11], where they factorized 20-bit integer $1,028,171$. Furthermore, general-purpose quantum computers have limited resources. The most potent Intel, IBM, and Google quantum computers have 49, 53, and 72 qubits, respectively [5], [10], [4]. Resources of general

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_12

quantum computers are nowadays too small to solve real-world cryptography problems.

This paper shows how to transform the relation searching step for the index calculus method on elliptic curves into the optimization problem. The relation searching problem is transformed into the QUBO (Quadratic Unconstrained Boolean Optimization) problem, where constraints are exchanged by penalties added to the objective function.

Even though Shor's quantum algorithm for factorization, discrete logarithm problem, and elliptic curves discrete logarithm problem is known to be efficient (it works in polynomial time), present quantum computers (even the most powerful) can solve ECDLP defined on at most 6-bit prime field $\mathbb{F}_p$. Using the index calculus method and transformation of relations searching step to the QUBO problem, we solved the biggest ECDLP problem using the quantum method nowadays. Using Leap hybrid sampler on the D-Wave Leap cloud, we broke ECDLP for elliptic curve defined over 8-bit prime field $\mathbb{F}_p$, where $p = 251$.

2 Index calculus method on elliptic curves using quantum annealing

2.1 Index calculus and summation polynomials

One of the most interesitng index calculus method on elliptic curve was firstly described by Semaev in [9], where he introduced summation polynomials. The summation polynomials have been defined there for elliptic curve in short Weierstrass form $E/\mathbb{F}_p : y^2 = x^3 + Ax + B$. Semaev summation polynomials have roots, when curve points sum to $\mathcal{O}$. The 2-nd Semaev polynomial is given by

$$f_3(x_1, x_2) = x_1 - x_2. \tag{1}$$

Using elementary methods it is also possible to find 3-rd Seamev polynomial as

$$\begin{aligned}&f_3(x_1, x_2, x_3) = x_2^2x_3^2 - 2x_1x_2x_3^2 + x_1^2x_3^2 - 2x_1x_2^2x_3 - 2x_1^2x_2x_3 \\ &-2Ax_2x_3 - 2Ax_1x_3 - 4Bx_3 + x_1^2x_2^2 - 2Ax_1x_2 - 4Bx_2 - 4Bx_1 + A^2.\end{aligned} \tag{2}$$

For any $m \geq 4$ and $m - 3 \geq k \geq 1$, one can find $f_m(x_1, \ldots, x_m)$ as

$$f_m(x_1, \ldots, x_m) = Res\left(f_{m-k}(x_1, \ldots, x_{m-k-1}, x), f_{k+2}(x_{m-k}, \ldots, x_m, x)\right). \tag{3}$$

$m+1$-th Summation polynomials$f_{m+1}(x_1, \ldots, x_m, x_R)$ is equal to zero iff there exist $y_1, \ldots, y_m$, for which every point of the form (x_i, y_i), where $i = \overline{1, m}$, lies on an elliptic curve and their sum $(x_1, y_1) + \cdots + (x_m, y_m)$ is equal to $(x_R, y_R) \in E(\overline{\mathbb{K}})$. Point (x_R, y_R) is computed as $[\alpha]P + [\beta]Q$ for some randomly chosen α and β. The roots of polynomial f_{m+1} should be found with a high probability if x_i is bounded by $p^{\frac{1}{m}}$.

2.2 Transformation of relation searching problem into the QUBO problem

Let us define the Semaev approach of relation searching

Problem 1

$$\begin{cases} f_{m+1}(x_1, \dots, x_m, x_R) = 0, \\ 0 \le x_1 \le p^{\frac{1}{m}}, \\ \dots \\ 0 \le x_m \le p^{\frac{1}{m}}, \end{cases} \tag{4}$$

where $f_{m+1}(x_1, \dots, x_m, x_R)$ is $m+1$-th Semaev polynomial.

We consider an approach of Semaev of relations searching given by Problem 1. To formulate the QUBO problem, we have to have some function to minimize, and we should not have any constraints. Problem 1 consists only of constraints. So to transform, one needs to take the following steps.

1. At first, all variables need to be changed to binary form. If variable z is from interval $U = \overline{a, b}$, the easiest way is to find surjection $g : 2^l \to U$, where $z = g(z_1, \dots z_l) = a + \sum_{i=0}^{s-1}(2^i z_i) + ((b-a)+1-2^s)z_l$, and $l = \lfloor \log_2(b-a) \rfloor$. This idea may be found in [2].
2. After transforming each of the variables, one has to make substitutions in polynomial f_{m+1}.
3. In the next step, we linearize the equation $f_{m+1}(x_1, \dots, x_m, x_R) = 0$, so all monomials of the degree of bigger than 1 have to be substituted using new variables $x_i x_j = u_k$. Additionally, these constraints also need to be changed to the penalty to add it to the QUBO problem, but penalties will be added later. Each penalty monomial of the form $x_i x_j x_l$ will be constructed, according to [6], in the following way $x_i x_j x_l \to u_k x_l = x_l u_k + 2(x_i x_j - 2u_k(x_i + x_j) + 3u_k)$.
4. In the next step, it is necessary to convert equation $f_{m+1}(x_1, \dots, x_m, x_R) = 0$, which is a modular equation, to equation over $\mathbb{Z}$. To make such transformation, it is necessary to write $f_{m+1}(x_1, \dots, x_m, x_R) - vp = 0$, where v is some positive integer. One can bound v if he previously computes the maximal value of f_{m+1} over $\mathbb{Z}$. Let us assume that maximal value of f_{m+1} over $\mathbb{Z}$ is equal to f_{max}. Then $v \le \log_2 \left\lfloor \frac{f_{max}}{p} \right\rfloor + 1$. Of course, we should also transform v into binary form, as presented in Step 1.
5. Finally, to use constraint that $f_{m+1}(x_1, \dots, x_m, x_R) - vp = 0$ in a minimization problem, one has to use this constraint as a penalty, where the right values of variables result in that such penalty is equal to 0, and the penalty is bigger than 0 otherwise. So one needs to construct this penalty as $(f_{m+1}(x_1, \dots, x_m, x_R) - vp)^2$. Because $f_{m+1}(x_1, \dots, x_m, x_R)$ was previously linearized, in $(f_{m+1}(x_1, \dots, x_m, x_R) - vp)^2$ only monomials of degree 2 may exist. Moreover, now one can add penalties obtained during linearization in the previous step. Each penalty is multiplied by a square of maximal coefficient appearing in $f_{m+1}(x_1, \dots, x_m, x_R)$, to get a high probability that minimal energy will be obtained only for a proper solution

After making the steps above, one obtains the QUBO problem, which has to be minimized.

Therefore, the number of variables in the QUBO problem is equal to $2\left(\left\lfloor \log_2 p^{\frac{1}{m}} \right\rfloor + 1\right) + \lfloor \log_2 v \rfloor + 1 + c$, where c is the number of auxiliary variables obtained during linearization.

Now we will estimate how many variables one needs to solve Problem 1 for $m = 2$. Let us note that (after transformation to binary form) f_3 is polynomial of degree 4 of $2l$ boolean variables, where $l = \left\lfloor \log_2 p^{\frac{1}{m}} \right\rfloor + 1$. It means that $\left(f_{m+1}(x_1, \ldots, x_m, x_R) - vp\right)^2$ is polynomial of degree 8 of $2\left(\left\lfloor \log_2 p^{\frac{1}{m}} \right\rfloor + 1\right) + \lfloor \log_2 v \rfloor + 1$ variables. Let us try to estimate the maximal value of v. Let us note that after transformation to boolean variables, in $f_{m+1}(x_1, \ldots, x_m, x_R)$ appear all possible combinations of monomials, where is one monomial of degree 0, $2l$ monomials of degree 1, $3l^2 - 2l$ monomials of degree 2, $2l(l^2 - l)$ monomials of degree 3, and $(l^2 - l)^2$ monomials of degree 4. Summing up, we have $(l^2 - l)^2 + 2l(l^2 - l) + (3l^2 - 2l) + 2l + 1 = l^4 + 2l^2 + 1 = (l^2 + 1)^2$ monomials at most. It means that $v_{max} = \left\lfloor \frac{(l^2+1)^2(p-1)}{p} \right\rfloor < \left(l^2 + 1\right)^2$.

For $m = 2$, if one wants to linearize polynomial $\left(f_{m+1}(x_1, \ldots, x_m, x_R) - vp\right)^2$ (penalties should be stored and added later, but there will not be necessary to obtain more auxiliary variables), then f_{m+1} will consist of at most $(l+1)^2$ variables. It means that $\left(f_{m+1}(x_1, \ldots, x_m, x_R) - vp\right)^2$ will require at most $(l+1)^2 + \left\lfloor \log_2 (l+1)^2 \right\rfloor + 1$ variables, because v is equal to at most $v_{max} = (l+1)^2$ and the bit length of v is equal to $\left\lfloor \log_2 (l+1)^2 \right\rfloor + 1$.

In the case of $m \geq 3$, we can estimate the total number of variables in the following way. At first, let us note that f_{m+1} is polynomial of $m+1$ variables, but in this case, the last variable (x_R) is fixed, so it means that we have m variables. For each variable in $m+1$-th Semaev polynomial, this polynomial is of degree 2^{m-1}. As previously, let us denote the bit length of each variable by l. Then, after transformation to binary form, f_{m+1} will consist of at most $\left(\sum_{i=0}^{2^{m-1}} l^i\right)^m$ monomials. If one wants to linearize this polynomial (penalties should be stored and added later, but there will not be necessary to obtain more auxiliary variables), then f_{m+1} will consist of at most $s = \left(\sum_{i=0}^{2^{m-1}} l^i\right)^m$ variables. It means that $\left(f_{m+1}(x_1, \ldots, x_m, x_R) - vp\right)^2$ will require at most $s + \lfloor \log_2 s \rfloor + 1$ variables, because v is equal to at most s and the bit length of v is equal to $\lfloor \log_2 s \rfloor + 1$.

Lagrange coefficient in penalties is equal to $2\left(Coeff_{max}\right)^2$, where $Coeff_{max}$ is the maximal coefficient of polynomial $f_{m+1}(x_1, \ldots, x_m, x_R)$. It is possible to obtain a QUBO problem in such a form that a minimal solution is always equal to 0, including offset. In such case, Lagrange coefficient of polynomial $f_{m+1}(x_1, \ldots, x_m, x_R)$ should be equal to $v_{max}p < \left(\sum_{i=0}^{2^{m-1}} l^i\right)^m p$. The disadvantage of this method is that one obtains larger coefficients in the QUBO problem. Therefore, it is harder to find the optimal solution because the minimal energy gap will be proportionally smaller.

Moreover, the QUBO problem may be (efficiently) solved on computers using quantum annealing like, for example, a D-Wave computer. An example of applying the method presented above for an elliptic curve $E/\mathbb{F}_{13} : y^2 = x^3 + 2x+4$, where $\#E(\mathbb{F}_p) = 17$ is presented in https://github.com/Michal-Wronski/ECDLP-index-calculus-using-QUBO/blob/main/QUBO_Example.pdf.

3 Experiments, results, and discussion

We analyzed an approach when the relation searching problem is transformed into the QUBO problem. We did several experiments for different sizes of p. In each experiment, we looked for a relationship for a given random point R. We used Magma Computational Algebra System (http://magma.maths.usyd.edu.au/magma/) for precomputations up to obtaining problems in the QUBO form. We used the D-Wave Leap cloud (cloud.dwavesys.com/leap/), which allows us to access the D-Wave computer remotely. Using this environment and D-Wave hybrid Leap sampler, we found a discrete logarithm on the elliptic curve $E/\mathbb{F}_p : y^2 = x^3 + Ax + B$, over 8-bit prime $p = 251$, where $A = 1, B = 4$. The curve E order is equal to $\#E(\mathbb{F}_p) = 271$, which is prime. The generator is equal to $P = (128, 44)$, and the resulting point is equal to $Q = [k]P = (95, 73)$. We used the previous section index calculus method applying the QUBO problem, with $m = 2$, to find discrete logarithm $\log_P Q = k = 157$. It is, of course, a tiny example. Still, one should note that according to [7] and [8], breaking ECDLP for the elliptic curve over an 8-bit prime would require 75 and 88 logical qubits, respectively, which is more than the biggest quantum computers nowadays have.

Using these estimations and number of variables, for $m = 2$, being not greater than $\left(l^2+1\right)^2 + \lfloor\log_2\left(l^2+1\right)^2\rfloor + 1$, we obtained that, using D-Wave, it would be (optimistically) possible to break ECDLP for 23-bit prime at most, interpreting our QUBO problem as dense (what seems to be more accurate in this case) and for 64-bit prime at most, interpreting our QUBO problem as general. Let us note that according to D-Wave Advantage documentation (https://www.dwavesys.com/d-wave-two-system), the maximal number of variables for dense problems is equal to $20,000$ and for general problems is equal to $1,000,0000$. One should also note that these are only theoretical expectations because the problem is minimal energy gap, which is proportionally smaller while p is growing. It results in that for larger p and larger m, the probability of obtaining minimal solution instead of suboptimal decreases.

Figure 1 presents how many variables obtained the QUBO problem require, depending on the bit length of p and given m. Let us note that for $m = 3$ and $m = 4$, the number of variables grows very fast. Thus, solving ECDLP using the index calculus method for these values of m (or larger) and D-Wave is impractical nowadays.

Furthermore, the size of the QUBO problem for relation searching, however, is polynomial to $\log_2 p$, but it grows very fast when m grows, and therefore, this approach seems to be impractical for solving real problems.

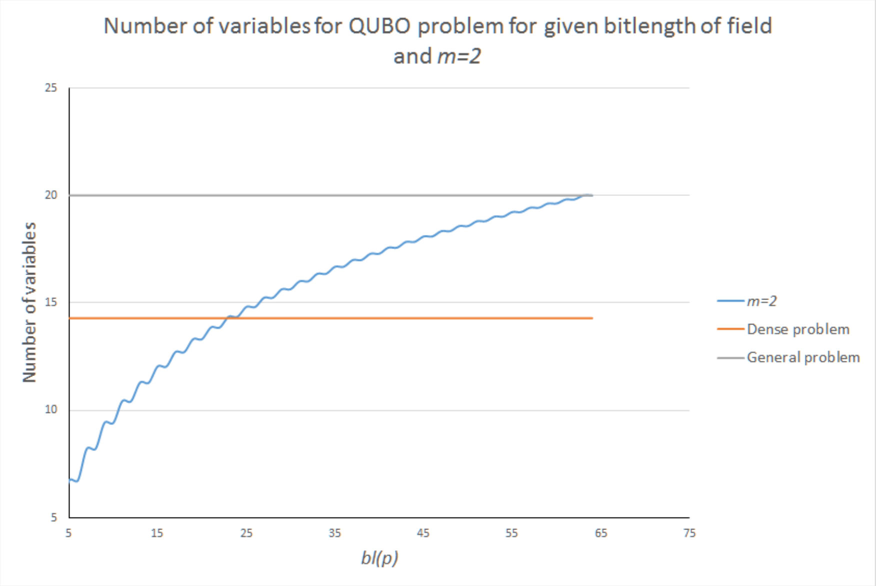

Fig. 1. Maximal total number of variables of QUBO problem for relation searching for $m = 2$ using D-Wave.

4 Conclusion and further works

Searching for a single relation depends on the computational complexity of solving a given optimization problem. It is possible to transform the relation searching problem into the QUBO problem. Solving the QUBO problem is exponential using classical algorithms [1], but the complexity of solving the QUBO problem using quantum annealing is unknown. Using the D-Wave Leap cloud, we broke ECDLP for the 8-bit prime field. Moreover, it would be possible to use D-Wave to break ECDLP for a maximally 23-bit prime field in a very optimistic case. However, it is still a small size problem compared to classical computers. It seems that it is far beyond the abilities of general-purpose quantum computers available nowadays.

The efficiency of the approach using QUBO may be increased by applying the following improvements:

- applying the different algorithm of quadratization of the polynomial $(f_{m+1}(x_1, \ldots, x_m, x_R) - vp)^2$, which would result in less number of total variables or smaller connectivity between variables,
- modifying a method of translation of relation searching step to the QUBO problem to obtain the larger value of minimal energy gap and thus decreasing the probability of obtaining suboptimal solution instead of the optimal solution,

- manual embedding of the given QUBO problem to the D-Wave Advantage computer. Using D-Wave hybrid Leap sampler, one cannot control how the problem is decomposed and if the solution is obtained classically or quantumly. Moreover, automatic embedding may give improper solutions.

It seems that if improvements above would be possible to apply, the index calculus method using QUBO would be much more efficient and could be solved on D-Wave for larger fields. Summing up, this approach seems to have potential and more research in this area should be done.

Acknowledgments

I would like to thank Christophe Petit for his apt comments and advice during the preparation of this paper.

References

1. Borle, A., Lomonaco, S.J.: Analyzing the quantum annealing approach for solving linear least squares problems. In: Das, G.K., Mandal, P.S., Mukhopadhyaya, K., Nakano, S.i. (eds.) WALCOM: Algorithms and Computation. pp. 289–301. Springer International Publishing, Cham (2019)
2. Chen, Y.A., Gao, X.S., Yuan, C.M.: Quantum algorithm for optimization and polynomial system solving over finite field and application to cryptanalysis. arXiv preprint arXiv:1802.03856 (2018)
3. Dridi, R., Alghassi, H.: Prime factorization using quantum annealing and computational algebraic geometry. Scientific reports **7**, 43048 (2017)
4. Greene, T.: Google reclaims quantum computer crown with 72 qubit processor (2018), https://thenextweb.com/artificial-intelligence/2018/03/06/google-reclaims-quantum-computer-crown-with-72-qubit-processor
5. Intel: The future of quantum computing is counted in qubits (2018), https://newsroom.intel.com/news/future-quantum-computing-counted-qubits/#gs.iiybkc
6. Jiang, S., Britt, K.A., McCaskey, A.J., Humble, T.S., Kais, S.: Quantum annealing for prime factorization. Scientific reports **8**(1), 1–9 (2018)
7. Proos, J., Zalka, C.: Shor's discrete logarithm quantum algorithm for elliptic curves. arXiv preprint quant-ph/0301141 (2003)
8. Roetteler, M., Naehrig, M., Svore, K.M., Lauter, K.: Quantum resource estimates for computing elliptic curve discrete logarithms. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 241–270. Springer (2017)
9. Semaev, I.A.: Summation polynomials and the discrete logarithm problem on elliptic curves. IACR Cryptol. ePrint Arch. **2004**, 31 (2004)
10. Shankland, S.: Ibm's new 53-qubit quantum computer is its biggest yet (2019), https://www.cnet.com/news/ibm-new-53-qubit-quantum-computer-is-its-biggest-yet/
11. Wang, B., Hu, F., Yao, H., Wang, C.: prime factorization algorithm based on parameter optimization of ising model. Scientific Reports **10**(1), 1–10 (2020)

DOI: 10.1007/978-3-030-77980-1_12