How Intractable Is the Discrete Logarithm for a General Finite Group?

Tatsuaki Okamoto* Kouichi Sakurai† Hiroki Shizuya‡

* NTT Laboratories
Nippon Telegraph and Telephone Corporation
1-2356, Take, Yokosuka-shi, Kanagawa-ken, 238-03 Japan
Email: okamoto@sucaba.ntt.jp

†Computer & Information Systems Laboratories
Mitsubishi Electric Corporation
5-1-1, Ofuna, Kamakura, 247 Japan
Email: sakurai@isl.melco.co.jp

‡Education Center for Information Processing
Tohoku University
Kawauchi, Aoba-ku, Sendai, 980 Japan
Email: shizuya@ecip.tohoku.ac.jp

Abstract

GDL is the discrete logarithm problem for a general finite group G. This paper gives a characterization for the intractability of GDL from the viewpoint of computational complexity theory. It is shown that GDL $\in$ NP $\cap$ co-AM, assuming that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in a polynomial time of the element size. Furthermore, as a natural probabilistic extension, the complexity of GDL is investigated under the assumption that the group law operation is executed in an expected polynomial time of the element size. In this case, it is shown that GDL $\in$ MA $\cap$ co-AM if $G \in$ NP $\cap$ co-NP. Finally, we show that GDL is less intractable than NP-complete problems unless the polynomial time hierarchy collapses to the second level.

R.A. Rueppel (Ed.): Advances in Cryptology - EUROCRYPT '92, LNCS 658, pp. 420-428, 1993.

1 Introduction

The discrete logarithm problem has played an important role in the construction of some cryptographic protocols. The problem is usually defined over the multiplicative group of a finite field, but it has some varieties with respect to its underlying finite group such as the multiplicative group over a finite ring modulo a composite, the elliptic curve group over a finite field [Mi, Ko1], the jacobian of the hyperelliptic curve over a finite field [Ko2], and so on. Since the computational complexity of each version of the discrete logarithm problem has some cryptographic or structural complexity-theoretic implications, it is important to characterize their complexity as well as to find efficient algorithms for solving them.

Our interest in this paper is to characterize the intractability of the general discrete logarithm problem that does not depend on a specific underlying finite group, from a viewpoint of structural complexity theory. The discrete logarithm problem for a general finite group G can be stated as follows: Given $a \in G$ and $b \in G$, find the smallest integer x such that $b = a^x$, provided that such an integer exists, where a^x denotes $a \circ a \circ \cdots \circ a$ (x times), and $\circ$ denotes the group law of G. The integer x is called the discrete logarithm of b to the base a.

Since the general discrete logarithm problem is a computing problem, we introduce a language GDL such that the complexity of membership problem in GDL is equivalent to that of the general discrete logarithm problem. Our goal is to show the class of the language GDL.

As an instance of GDL, when G is a multiplicative group over a finite field, we call GDL "MDL". When G is a hyperelliptic discrete logarithm [SIS], we call GDL "HEDL". Brassard has pointed out that MDL $\in$ NP $\cap$ co-NP [Br]. Shizuya, Itoh and Sakurai showed that HEDL $\in$ NP $\cap$ co-AM [SIS], after which Okamoto and Sakurai showed that HEDL $\in$ NP $\cap$ co-NP, provided that the jacobian of the hyperelliptic curve is non-half-degenerate [OS], where, for example, the jacobians with the most general (complicated) structure satisfy this condition (or non-half-degeneracy), and any jacobian with genus one (elliptic curve) also satisfies this condition. However, it has not been shown which class GDL belongs to, if G is a general finite group.

In this paper, we show that the result for HEDL by [SIS] can be generalized to GDL for a general finite group. That is, we show that GDL $\in$ NP $\cap$ co-AM, assuming that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in a polynomial time of the element size. If the group law operation is executed in an expected polynomial time and $G \in$ NP $\cap$ co-NP, we show that

GDL $\in$ MA $\cap$ co-AM.

2 Preliminaries

Throughout this paper, all strings will be over the finite alphabet $\Sigma = \{0, 1\}$. We use $|x|$ to represent the length of string x. We let Σ^* designate the set of all possible strings including zero-length string λ. A language is a set of strings. A class is a set of languages. For a language L, we use co-L to denote $\Sigma^* \setminus L$. For a class $\mathcal{C}$, we use co-$\mathcal{C}$ to denote its complement, i.e. the set of any L such that co-L is in $\mathcal{C}$.

We define the following language GDL to investigate the complexity of the general discrete logarithm problem for a general finite group G. It is clear that the complexity of the general discrete logarithm problem is equivalent to that of the membership problem in GDL.

Definition 2.1 *Let $\mathcal{N}$ be a countably infinite set, and let $N \in \mathcal{N}$ specify a finite group G_N. Define $G = \{G_N \mid N \in \mathcal{N}\}$. Given N and $a \in G_N$, a^x is calculated by the group law of G_N.*

$$\text{GDL} = \{(a, b, N, k) \mid a, b \in G_N \wedge k \in Z \wedge \exists x[b = a^x \wedge 0 \leq x \leq k]\}.$$

Here, we assume that, given N, for any $a, b \in G_N$, $|a| = |b|$. [1]

Furthermore, we take into account the complexity of the decision problem that, given N and x, asks whether x is in G_N or not. Such a decision is often required to check the validity of input strings, or the correctness of output strings in some computation process. For this purpose, we alternatively regard the general group G as a language over Σ^*, and will sometimes say that, for example, G is in P.

We will assume in subsequent sections that G is in P or in NP $\cap$ co-NP because it is reasonable to consider only finite groups such that there is a witness for $x \in G$ or $x \notin G$. (Another research topics would be to extend the discussion to cover groups out of this assumption, for example, G in a class beyond NP or in some probabilistic class.) We will also assume that there exists a program which calculates the group law, runs in a (deterministic or expected) polynomial time, and outputs only a correct answer.

Example 1: Let Z_p be a finite prime field of characteristic p, and let Z_p^* be its multiplicative group. Suppose $G = \{Z_p^*\}$ (or $G_p = Z_p^*$), which is true in the

[1] From this assumption, when (a, b, N, k) is an input of GDL, $|\text{ord}(a)|$ can be bounded by $|a|$ or the input size, $|(a, b, N, k)|$.

case of MDL. Then, it is known that MDL is in NP ∩ co-NP [Br]. Here, the multiplication over Z_p can be executed in deterministic polynomial time in $|p|$. G is clearly in P because given x, we can immediately check that x is a positive integer less than p.

Example 2: Let $E(C, F_q)$ be an elliptic curve group over a finite field of characteristic p, where C is the equation that gives the curve, and $q = p^n$. Suppose $G = \{E(C, F_q)\}$ (or $G_{(C,q)} = E(C, F_q)$), which is the special case of HEDL called EDL. Then, it is known that EDL is in NP ∩ co-NP [SIS, OS]. Here, the addition of two (possibly distinct) points on the elliptic curve can be executed in deterministic polynomial time. G is in P because given a point Q, we can check in deterministic polynomial time in $|p|$ that Q satisfies C over F_q.

Example 3: Let Z_n be a finite residue class ring modulo n, and let QR_n be the group of quadratic residues over Z_n. Suppose $G = \{QR_n\}$ (or $G_n = QR_n$). The group law is then simply the multiplication over Z_n, and it can be executed in deterministic polynomial time in $|n|$. However, unlike other examples, G is in NP ∩ co-NP, known as quadratic residuosity modulo n. In this case, we must take into account the complexity of G in order to characterize the complexity of the discrete logarithm problem over QR_n because checking the validity of input strings contains an NP-statement rather than an easily decidable P-statement. This is a typical example that indicates why we consider the complexity of G in our characterization for GDL.

3 Main Result

Theorem 3.1 *GDL ∈ NP ∩ co-AM, assuming that G is in P, and that the group law operation of G can be executed in a deterministic polynomial time of the element size.*

Proof: It is trivial that GDL ∈ NP, since the witness of GDL is x such that $b = a^x \wedge x \leq k$. To show that GDL ∈ co-AM, it is sufficient to show that co-GDL ∈ AM. There are two cases when $(a, b, N, k) \notin$ GDL. One is the case that there exists x such that $b = a^x$ and that $k < x < \mathrm{ord}(a)$. The other is the case that there does not exist x such that $b = a^x$. In the former case, co-GDL ∈ NP ⊆ AM, since x, ord(a) and all prime factors of ord(a) are the witness of $(a, b, N, k) \notin$ GDL.

Hence, in the remaining part of this proof, we will show that co-GDL ∈ AM in the latter case. To show that there does not exist x such that $b = a^x$, it is sufficient to show that $b \notin < a >$, where $< a >$ denotes the subgroup generated

by a.

The following protocol is the constant round interactive proof system that shows $b \notin < a >$. Combining this and the result by [GS], we can conclude that co-GDL $\in$ AM in the latter case. Thus, GDL $\in$ NP $\cap$ co-AM.

Protocol:

Step 1 Prover P computes t (the order of a), $(s_1, \ldots, s_k)$ (all prime factors of t), and $(u_1, \ldots, u_k)$ (the witnesses of the primility of $s_1, \ldots, s_k$ [Pr]). P sends them to verifier V.

Step 2 V checks the correctness of t, by checking $a^t = e$ and $a^{t/s_i} \neq e$ for any $i = 1, \ldots, k$. V also checks the correctness of the primality of $(s_1, \ldots, s_k)$ using $(u_1, \ldots, u_k)$ through Pratt's algorithm [Pr]. If it is not correct, V halts. Otherwise, V selects a random bit $c \in \{0, 1\}$ and a random integer $r \in Z_t$. Then, V computes $d = a^r \circ b^c$. V sends d to P.

Step 3 P sets $c' = 0$ if $d \in < a >$, and sets $c' = 1$ if $d \notin < a >$. P sends c' to V.

Step 4 V checks whether $c = c'$. If $c \neq c'$, V rejects and halts. Otherwise, V continues the protocol.

Step 5 After the protocol above is repeated in a constant round, V accepts the proof if $c = c'$ for all rounds.[2].

Finally, we show that the above protocol is an interactive proof system for $b \notin < a >$.

(Completeness:) When $b \notin < a >$, assume that $a^r \circ b \in < a >$. Then, there exists l such that $a^r \circ b = a^l$. Hence, $b = a^{l-r}$. This is a contradiction. Therefore, if $b \notin < a >$, then $a^r \circ b \notin < a >$. Thus, when $b \notin < a >$, $d \in < a >$ if $c = 0$, and $d \notin < a >$ if $c = 1$. That is, $c = c'$ for all rounds. Thus, when $b \notin < a >$, V accepts the proof with probability 1.

(Soundness:) If $b \in < a >$ or $b = a^u$, then $a^r \circ b$ distributes uniformly over $< a >$, since $a^r \circ b = a^{r+u}$ and $r + u \bmod t$ distributes uniformly over Z_t. Clearly, a^r distributes uniformly over $< a >$. Therefore, when $b \in < a >$, no P' can guess the value of c with a probability of more than $1/2$. Thus, the probability that P' convinces V with a constant ($K > 1$) round repetition is at most $1/2^K < 1/3$.

□

[2] This procedure can be parallelized

Since we assume in the above that G is in P, we do not explicitly consider the validity check of strings. Whereas, if G is not known to be in P, it is necessary to take into account the complexity of G as shown in Example 3. However, the following theorem shows that the complexity of GDL is *not* affected if G is in NP $\cap$ co-NP.

Corollary 3.2 *GDL $\in$ NP $\cap$ co-AM, assuming that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in a deterministic polynomial time of the element size.*

Proof: The strategy of our proof is the same as for the previous theorem except that we use G as an oracle set to check the validity of input strings or the correctness of output strings in some computation process.

It is clear that GDL is recognized by a polynomial time bounded nondeterministic oracle Turing machine with G oracle. Since G is in NP $\cap$ co-NP, GDL is at most in $\mathrm{NP}^{\mathrm{NP}\cap\mathrm{co-NP}}$. However, by the result on low and high hierarchies within NP [Sch], $\mathrm{NP}^{\mathrm{NP}\cap\mathrm{co-NP}} = \mathrm{NP}$. Thus, GDL $\in$ NP.

To prove GDL $\in$ co-AM, we can show almost the same constant round interactive protocol for $b \notin < a >$. The only one difference is that the verifier V is allowed to make queries to the oracle set G. Thus, co-GDL is at most in $\mathrm{AM}^{\mathrm{NP}\cap\mathrm{co-NP}}$. However, by the result on lowness in probabilistic complexity classes [Kl], $\mathrm{AM}^{\mathrm{AM}\cap\mathrm{co-AM}} = \mathrm{AM}$. Thus, co-GDL $\in$ AM, and we conclude that GDL $\in$ NP $\cap$ co-AM. □

The following corollaries are the consequences of natural probabilistic extension for the group law operation.

Corollary 3.3 *GDL $\in$ MA $\cap$ co-AM, assuming that G is in P, and that the group law operation of G can be executed in an expected polynomial time of the element size.*

Proof: ¿From the assumption, the operation of a^x can be executed in expected polynomial time. Therefore, GDL $\in$ MA, since the prover (Merlin, M) sends the verifier (Arthur, A) the witness of GDL, x ($b = a^x \ \wedge \ x \leq k$), and a polynomial-time machine, A, can check the correctness of x with a probablity of more than 2/3.

Similarly, to show that GDL $\in$ co-AM, it is sufficient to show that co-GDL $\in$ MA when there exists x such that $b = a^x$ and that $x > k$, and to show that co-GDL $\in$ AM when there does not exist x such that $b = a^x$. In the former case, clearly co-GDL $\in$ MA $\subseteq$ AM. In the latter case, co-GDL $\in$ AM, since a

modification of the protocol shown in the proof of Theorem 2.1 can become the constant round IP to show that $b \notin < a >$, where in Step 5 V accepts the proof if $c = c'$ for 2/3 of all rounds. □

Corollary 3.4 *GDL $\in$ MA $\cap$ co-AM, assuming that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in an expected polynomial time of the element size.*

Proof: It suffices to show that GDL is in MA under these assumptions. In the MA protocol in the previous corollary, we allow Arthur (A) to make queries to the oracle set G. Since G is in NP $\cap$ co-NP, GDL is at most in $\mathrm{MA}^{\mathrm{NP}\cap\mathrm{co-NP}}$. Note in the proof that it is not known whether $\mathrm{MA}^{\mathrm{NP}\cap\mathrm{co-NP}} = \mathrm{MA}$. However, since Arthur's query is made only once in order to check the validity of input strings from the assumption, GDL still remains in MA. Because, by the robustness of NP, the response to the query can be merged with Merlin's strings and sent to Arthur. □

The following result is obtained directly from the above results and the result by [BHZ]. This result implies that, for any complicated finite group G (under the reasonable assumption), GDL cannot be so intractable as NP-complete problems unless the polynomial time hierarchy collapses to the second level.

Corollary 3.5 *Assume that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in an expected polynomial time of the element size. Then, if GDL $\in$ NP-complete, the polynomial time hierarchy collapses to the second level.*

4 Related Open Problems

An open question is to find an instance of the finite group G such that (i) G is not known to be in P but is in NP $\cap$ co-NP, and (ii) a deterministic polynomial time algorithm to compute its group law operation is not known to exist, but an expected polynomial time algorithm is available. We let GDL* designate GDL for such a group. The complexity of GDL* is characterized as MA $\cap$ co-AM, which contrasts to the fact that MDL, GDL for a multiplicative group over a finite field, and HEDL, GDL for the hyperelliptic discrete logarithm problem, are both in NP $\cap$ co-NP [Br, OS].

A related question arises as to whether GDL* is in SZK, the class of languages that have statistical zero-knowledge interactive proof systems. It is known that

both MDL and HEDL have perfect zero-knowledge interactive proof systems, respectively [TW, SIS]. By the results of [AH, Fo], SZK $\subseteq$ AM $\cap$ co-AM, but it is not known whether the converse holds.

5 Conclusion

In this paper, we have shown that GDL $\in$ NP $\cap$ co-AM, assuming that G is in NP $\cap$ co-NP, and that the group law operation of G can be executed in a polynomial time of the element size. We extended the discussion to the case where the group law operation is executed in an expected polynomial time of the element size, and have shown that GDL is in MA $\cap$ co-AM if G is in NP $\cap$ co-NP. Finally we have shown that GDL cannot be NP-complete unless the polynomial time hierarchy collapses to the second level.

Acknowledgments

Authors wish to thank Claude Crépeau for his initial suggestion which led to this paper. They would also like to thank Birgit Pfitzmann and Osamu Watanabe for their invaluable comments and discussions on the preliminary version. They also thank Lance Fortnow for informing the correctness of Klapper's result on $\mathrm{AM}^{\mathrm{AM}\cap\mathrm{co-AM}} = \mathrm{AM}$. The third author would like to thank Gilles Brassard for his encouragement on this work. They thank anonymous referees for their useful comments.

References

[AH] W. Aiello, J. Håstad, "Perfect zero-knowledge languages can be recognized in two rounds," Proc. 28th FOCS, pp.439-448 (1987).

[BHZ] R. Boppana, J. Håstad, and S. Zachos, "Does co-NP have short interactive proofs?" Inform. Proc. Lett., vol.25, pp.127-132 (1987).

[Br] G. Brassard, "A note on the complexity of cryptography," IEEE Trans. Inf. Theory, vol.IT-25, no.2, pp.232-233 (1979).

[Fo] L. J. Fortnow, "The complexity of perfect zero-knowledge," Proc. 19th STOC, pp.204-209 (1987).

[GS] S. Goldwasser and M. Sipser, "Private coins versus public coins in interactive proof systems," Proc. 18th STOC, pp.59-68 (1986).

[Kl] A. Klapper, "Generalized lowness and highness and probabilistic complexity classes," Math. Syst. Theory, vol.22, pp.37-45 (1989).

[Ko1] N. Koblitz, "Elliptic curve cryptosystems," Math. Comp., vol.48, no.177, pp.203-209 (1987).

[Ko2] N. Koblitz, "Hyperelliptic cryptosystems," J. Cryptology, vol.1, no.3, pp.139-150 (1989).

[Mi] V. S. Miller, "Use of elliptic curves in cryptography," *Advances in Cryptology: Proceedings of Crypto'85*, LNCS 218, pp.417-426 (1985).

[OS] T. Okamoto, K. Sakurai, "Efficient algorithms for the construction of hyperelliptic cryptosystems," *Advances in Cryptology: Proceedings of Crypto'91*, LNCS 576, Springer-Verlag, pp.265-278 (1992).

[Pr] V. R. Pratt, "Every prime has a succinct certificate," SIAM J. Comput., vol.4, pp.214-220 (1975).

[Sch] U. Schöning, "A low and high hierarchy within NP," J. Comp. Syst. Sci., vol.27, pp.14-28 (1983).

[SIS] H. Shizuya, T. Itoh, K. Sakurai, "On the complexity of hyperelliptic discrete logarithm problem", *Advances in Cryptology: Proceedings of Eurocrypt'91*, LNCS 547, Springer-Verlag, pp.337-351 (1991).

[TW] M. Tompa and H. Woll, "Random self-reducibility and zero knowledge interactive proofs of possession of information," Proc. 28th FOCS, pp.472-482 (1987).