

# Classical and Quantum Algorithms for Exponential Congruences

Wim van Dam

and

Igor E. Shparlinski

University of California, Santa Barbara

and

Macquarie University, Sydney

# Introduction

## Equation

$\mathbb{F}_q$  = finite field of  $q$  elements

$\mathbb{F}_q^*$  = multiplicative group of  $\mathbb{F}_q$ .

For  $a, b, c, f, g \in \mathbb{F}_q^*$  we consider the equations

$$af^x + bg^y = c \quad (1)$$

in nonnegative integers  $x$  and  $y$ .

## Motivation

*D. Bacon, A.M. Childs and W. van Dam, 2005:*

(1) is of importance when trying to solve the *hidden subgroup problem* for semi-direct product groups  $\mathbb{Z}/N \rtimes \mathbb{Z}/p$  with  $p = \Theta(\sqrt{N})$

*A. Lenstra, B de Weger, 2005:*

(1) is of cryptographic significance

It also appears in the theory *cyclotomic classes* and is a natural generalization of the discrete logarithm problem.

2

## Our Results

We use some number theoretic results, to design

[classical and quantum algorithms](#)

that are more efficient than the brute force search (but unfortunately still exponential in  $\log q$ ).

We use our classical algorithm to measure the level of improvement that can be achieved by allowing quantum algorithms.

In particular, it gives an example of a natural problem where quantum algorithms provide an asymptotically **cubic** speed-up over classical ones.

3

## Easy Case

If  $f$  or  $g$  is a primitive root, then the problem is not harder than the DLOG problem. In general our results suggest that finding solutions to Equation (1) becomes easier in cases where  $f$  or  $g$  is of large order, but still it appears to be harder than the DLOG problem.

4

# Algorithms

## Classical Algorithms:

We start with a classical deterministic algorithm that is more efficient than brute search.

**Theorem 1** *Let  $a, b, c, f, g \in \mathbb{F}_q^*$ . One can either find a solution  $x, y \in \mathbb{Z}_{\geq 0}$  of the equation  $af^x + bg^y = c$  or decide that it does not have a solution in deterministic time  $q^{9/8}(\log q)^{O(1)}$  on a classical computer.*

5

Furthermore, for almost all  $c$  a stronger result holds.

**Theorem 2** *Let  $a, b, c, f, g \in \mathbb{F}_q^*$ . For all but  $o(q)$  elements  $c \in \mathbb{F}_q^*$ , one can either find a solution  $x, y \in \mathbb{Z}_{\geq 0}$  of the equation  $af^x + bg^y = c$  or decide that it does not have a solution in deterministic time  $q(\log q)^{O(1)}$  on a classical computer.*

## Quantum Algorithms:

On a quantum computer one has the advantage that calculating discrete logarithms can be done efficiently. In combination with the quadratic speed-up of quantum searching this gives the following quantum algorithm for the central problem.

We start with an algorithm that works for *any*  $f$  and  $g$ .

**Theorem 3** *Let  $a, b, c, f, g \in \mathbb{F}_q^*$  and let  $f$  and  $g$  be of multiplicative orders  $s$  and  $t$ , respectively. There is an absolute constant  $C$  such that, on a quantum computer, one can either find a solution  $x, y \in \mathbb{Z}_{\geq 0}$  of the equation  $af^x + bg^y = c$  or decide that it does not have a solution in time  $q^{3/8}(\log q)^{O(1)}$  for any  $s$  and  $t$ ; and in time  $q^{1/2}(st)^{-1/4}(\log q)^{O(1)}$  if  $st > Cq^{3/2}(\log q)^{1/2}$ .*

7

As in the classical case, for almost all  $c \in \mathbb{F}_q$  stronger results are possible.

**Theorem 4** *Let  $a, b, c, f, g \in \mathbb{F}_q^*$  and let  $f$  and  $g$  be of multiplicative orders  $s$  and  $t$ , respectively. On a quantum computer, for all but  $o(q)$  elements  $c \in \mathbb{F}_q^*$ , one can either find a solution  $x, y \in \mathbb{Z}_{\geq 0}$  of the equation  $af^x + bg^y = c$  or decide that it does not have a solution in time  $q^{1/3}(\log q)^{O(1)}$  for any  $s$  and  $t$ ; and in time  $q^{1/2}(st)^{-1/4}(\log q)^{O(1)}$  if  $st > q^{4/3}(\log q)^{2/3}$ .*

# Connection with the Hidden Subgroup Problem and Open Problems

*D. Bacon, A.M. Childs and W. van Dam,*  
**2005:**

The Hidden Subgroup Problem over the group  $\mathbb{Z}_q \rtimes \mathbb{Z}_p$  with  $q$  a prime and  $q/p^2 = (\log q)^{O(1)}$  shows that this problem can be solved efficiently on a quantum computer if one can efficiently solve the equation

$$af^x + bf^y = c$$

where  $f$  has multiplicative order  $p$  in  $\mathbb{Z}_q$ .

Unfortunately, all algorithms presented in this paper fall short of this goal.

For this restricted problem with  $f = g$  and  $f$  of order  $p \approx \sqrt{q}$ , there are  $p^2$  possible solutions  $(x, y)$ , hence even a classical algorithm has  $\tilde{O}(q)$  time complexity instead of the  $\tilde{O}(q^{9/8})$  of Theorem 1.

Quantum mechanically, one can ‘Grover search’ the set of solutions  $x \in \{0, \dots, p - 1\}$  in time  $\tilde{O}(q^{1/4})$ , which is better than the  $\tilde{O}(q^{3/8})$  of Theorem 3 but is still far from polynomial in  $\log q$ .

**Open Question** Do there exist quantum algorithms that run in time  $(\log q)^{O(1)}$  for solving the equation  $af^x + bg^y = c$  and the more restricted version  $af^x + bf^y = c$  over  $\mathbb{F}_q$ .

10

More efficient classical algorithms?

In some finite fields classical subexponential probabilistic algorithms are possible for the DLOG problem.

In such fields, a version of Theorem 1 can be obtained with an algorithm that runs in probabilistic time  $q^{3/4+o(1)}$ , which is still much slower than the quantum algorithm of Theorems 3 and 4.

One can obtain analogues of our results in the elliptic curve setting, where no subexponential algorithms for the DLOG problem are known/expected.

# Proofs

## Idea

- Find the orders  $s$  and  $t$  of  $f$  and  $g$ , respectively
- If one of them (say,  $t$ ) is small, then we search through all  $y = 1, \dots, t$  and try to find the DLOG of  $(bg^y - c)/a$  to base  $f$ . Since  $t$  is small this is not too expensive.
- If both of them are large, we show that there is always a solution to (1) with  $y \leq r$  for some reasonably small  $r$ . We now proceed as in in the previous case but run  $x$  up to  $r$ .

**Obtaining a good estimate on  $r$  is crucial!**

## Technology

To estimate  $r$ , we use traditional number theoretic tools such as bounds of multiplicative and additive character sums such as

$$\sum_{y=0}^{r-1} \chi(\alpha g^y + \beta) \quad \text{and} \quad \sum_{y=0}^{r-1} \psi(\alpha g^y)$$

where  $\chi$  and  $\psi$  are multiplicative and additive characters, respectively.

In fact we obtain an asymptotic formula for the number of solutions to (1) with  $y \leq r$  which has an additional advantage in the quantum case as we can promise a certain density of solutions.

## Fun Part - Details

### Distribution of Solutions

**Lemma 5** *Let  $a, b, c \in \mathbb{F}_q^*$  and let  $f$  and  $g \in \mathbb{F}_q$  be of multiplicative orders  $s$  and  $t$ , respectively. Then for any positive integer  $r \leq t$ , the equation (1) has*

$$N_{a,b,c}(r, s) = \frac{rs}{q-1} + O(q^{1/2} \log q)$$

*solutions in nonnegative integers  $x$  and  $y$  with  $x \in \{0, \dots, s-1\}$  and  $y \in \{0, \dots, r-1\}$ .*

### Proof

Put  $k = (q-1)/s$

Let  $\mathcal{X}_k = \{\chi : \chi^k = \chi_0\}$  be the group of all  $k$  multiplicative characters  $\chi$  of order  $k$ ,

Orthogonality:

$$\frac{1}{k} \sum_{\chi \in \mathcal{X}_k} \chi(u) = \begin{cases} 1, & \text{if } u^s = 1, \\ 0, & \text{otherwise,} \end{cases}$$

$$u \in \langle f \rangle \quad \text{iff} \quad u^s = 1$$

$$\Downarrow$$

$$N_{a,b,c}(r, s) = \sum_{y=0}^{r-1} \frac{1}{k} \sum_{\chi \in \mathcal{X}_k} \chi(a^{-1}(c - bg^y)).$$

Changing the order of summation and separating the term  $r/k$  corresponding to the principal character  $\chi_0$  we obtain

$$\left| N_{a,b,c}(r, s) - \frac{r}{k} \right| \leq \frac{1}{k} \sum_{\chi \in \mathcal{X}_k \setminus \{\chi_0\}} \chi(a^{-1}) \sum_{y=0}^{r-1} \chi(c - bg^y).$$

*E. Dobrowolski and K. S. Williams, 1992:*

*H.B. Yu, 2001:*

$$\sum_{y=0}^{r-1} \chi(c - bg^y) = O(q^{1/2} \log q).$$

Hence

$$N_{a,b,c}(r, s) = \frac{r}{k} + O(q^{1/2} \log q),$$

which concludes the proof.

**Corollary 6** *Let  $a, b, c \in \mathbb{F}_q^*$  and let  $f$  and  $g$  be of multiplicative orders  $s$  and  $t$ , respectively. There exists an absolute constant  $C > 0$  such that if for some integer  $r$  we have*

$$Cq^{3/2}s^{-1} \log q \leq r \leq t,$$

*then the equation  $af^x + bg^y = c$  has a solution in integers  $x$  and  $y$  with  $x \in \{0, \dots, s-1\}$  and  $y \in \{0, \dots, r-1\}$ .*

## Algorithm

We recall **Theorem 3**

*Let  $a, b, c, f, g \in \mathbb{F}_q^*$ . One can either find a solution  $x, y \in \mathbb{Z}_{\geq 0}$  of the equation  $af^x + bg^y = c$  or decide that it does not have a solution in time  $q^{3/8}(\log q)^{O(1)}$  on a quantum computer.*

### Proof

Using Shor's discrete logarithm algorithm:

- Create a poly-time quantum subroutine  $\mathcal{S}(x)$  that, for a given  $x$  either finds and returns the integer  $y$  with  $g^y = b^{-1}(c - af^x)$  or reports that no such  $y$  exists.
- Create a quantum subroutine  $\mathcal{T}(y)$  that, for a given  $y$  either finds and returns the integer  $x$  with  $f^x = a^{-1}(c - bg^y)$  or reports that no such  $x$  exists.

17

Use Shor's algorithm to compute  $s$  and  $t$ .

We assume that  $s \geq t$  and define

$$r = \lceil Cq^{3/2}s^{-1} \log q \rceil.$$

- If  $r \leq t$ , then using Grover's algorithm, search the subroutines  $\mathcal{S}(x)$  for all  $x \in \{0, 1, \dots, r - 1\}$  in time

$$\begin{aligned} r^{1/2}(\log q)^{O(1)} &= q^{3/4}s^{-1/2}(\log q)^{O(1)} \\ &\leq q^{3/8}(\log q)^{O(1)}. \end{aligned}$$

Due to our choice of  $r$ , by Corollary 6 in this case there is always a solution.

- If  $r > t$ , then we search the subroutines  $\mathcal{T}(y)$  for all  $y \in \{0, 1, \dots, t - 1\}$ , in time

$$\begin{aligned} t^{1/2}(\log q)^{O(1)} &= (st)^{1/4}(\log q)^{O(1)} \\ &\leq q^{3/8}(\log q)^{O(1)}. \end{aligned}$$

We either find a solution or conclude that there is no solution.