LA-UR 96-3793

CONF-9611146--1

TITLE: DETERMINING CONSTANTS IN SHOR'S DISCRETE LOG ALGORITHM OVER Zp

AUTHOR(S): J. MARK ETTINGER, NIS-9
MIKE NEEGAARD, NIS-9

SUBMITTED TO:

Los Alamos Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO 836 R4
ST NO 2629 5/81

Determining Constants in Shor's Discrete Log Algorithm over Z_p

J. Mark Ettinger [*]and Mike Neergaard [†]
Los Alamos National Laboratory

We compute the number of gates and laser pulses required in Shor's algorithm for finding r in the equation $g^r = x(mod\ p)$ where p is a prime, and g and x are given elements of the multiplicative group Z_p^*. Because the discrete logarithm algorithm is similar to the factoring algorithm we may utilize gates designed for use in factoring. We use the gate configurations and pulse counts found in [1] and [2]. We use the notation of [2] in which $[a, b, c, d, e]$ indicates a use of a Not gates, b controlled-Not gates, c controlled-controlled-Not gates, d controlled-controlled-controlled-Not gates, and e controlled-controlled-controlled-controlled-Not gates. Similarly $[a, b, c]$ will indicate an analogous gate structure where the control bits are limited to 2 controls (as in the *basic machine model* to be discussed below). Recall a k-controlled-Not gate requires $2k + 3$ pulses.

Let p be an L-bit prime. We first prepare the state

$$\frac{1}{2^L - 1} \sum_{a,b=0}^{2^L-1} |a, b>$$

which requires $2L$ qubits for register storage and one pulse for each bit. We next prepare the state

$$\sum_{a,b=0}^{2^L-1} |a, b, S(a), S(b)>$$

[*]ettinger@lanl.gov
[†]dude@lanl.gov

where $S(x) = 0$ if $x \leq p-2$ and $S(x) = 1$ otherwise. The S comparison function may be computed by reversing the plain adder network in [1]. This gate requires $[0, 4L-1, 4L-2]$, $48L-19$ pulses, and two qubits to store the comparison results. We observe these latter two bits and restart the entire process if we obtain a 1 in either bit. If we observe 0 in both registers we then know that we have created the state

$$\frac{1}{p-1}\sum_{a=0}^{p-2}\sum_{b=0}^{p-2}|a, b> .$$

To this point we have used $2L+2$ qubits, $50L-19$ pulses, and a gate structure of $[0, 4L-1, 4L-2]$. The next state we create is

$$\frac{1}{p-1}\sum_{a=0}^{p-2}\sum_{b=0}^{p-2}|a, b, g^a x^{-b}(mod\, p)> .$$

The requirements for modular exponentiation are varied according to time-space tradeoffs and whether or not we permit 3-controlled-Not gates and 4-controlled-Not gates in our machine (enhanced machine model) or restrict the control bit structure to 2-controlled-Not (basic machine model). We summarize here the results listed in [2] for convenience. The pairs of equations below give the gate structure and pulse count for the *average case complexity* for exponentiation modulo p, denoted $[EXPP]$, where the number of scratch bits is listed in the subscript. For example, the first equation gives the gate count for exponentiation modulo p assumimg the enhanced machine model and $2L+1$ qubits of scratch space and is therefore denoted $[EXPP]^{ave\ gates}_{enhanced,2L+1}$. The second equation gives the pulse count for exponentiation mod p assuming the enhanced machine model and $2L+1$ qubits of scratch space and is therefore denoted $[EXPP]^{ave\ pulses}_{enhanced,2L+1}$.

$$[EXPP]^{ave\ gates}_{enhanced,2L+1} = (L-1)[10L^2-14L+4, 4L^2+8L-12, 17L^2-36L+22, 3L^2-3, 2L^2-4L+2]$$
$$+[2, L/2+1, 0, 0, 0]$$
$$[EXPP]^{ave\ pulses}_{enhanced,2L+1} = (L-1)(198L^2-270L+93)+5L/2+7$$
$$[EXPP]^{ave\ gates}_{enhanced,2L+2} = (L-1)[10L^2-14L+4, 5L^2+10L-14, 19L^2-34L+21, 2L^2-4L+2, 0]$$
$$+[2, L/2+1, 0, 0, 0]$$

$$[EXPP]^{ave\ pulses}_{enhanced,2L+2} = (L-1)(186L^2 - 238L + 99) + 5L/2 + 7$$
$$[EXPP]^{ave\ gates}_{basic,2L+3} = (L-1)[10L^2-14L+4, 7L^2+6L-12, 23L^2-42L+25]+[2, L/2+1, 0]$$
$$[EXPP]^{ave\ pulses}_{basic,2L+3} = (L-1)(206L^2 - 278L + 119) + 5L/2 + 7$$
$$[EXPP]^{ave\ gates}_{basic,2L+2} = (L-1)[10L^2-14L+4, 5L^2+10L-14, 27L^2-50L+29]+[2, L/2+1, 0]$$
$$[EXPP]^{ave\ pulses}_{basic,2L+2} = (L-1)(224L^2 - 314L + 137) + 5L/2 + 7$$
$$[EXPP]^{ave\ gates}_{basic,2L+1} = (L-1)[10L^2-14L+4, 4L^2+8L-12, 49L^2-76L+30]+[2, L/2+1, 0]$$
$$[EXPP]^{ave\ pulses}_{basic,2L+1} = (L-1)(373L^2 - 506L + 154) + 5L/2 + 7$$

Notice that to produce the desired state it seems we must do *two* modular exponentiations and then multiply the results. A more efficient technique involves modifying the standard exponentiation circuit, which is a series of conditional multiplications, to produce a circuit that directly produces $g^a x^{-b}(mod\ p)$ on input a and b by cascading *all* the conditional multiplications from both exponentiation calculations. In other words (using the notation from [1] we calculate

$$g^a x^{-b}(mod\ p) = g^{\sum_i a_i 2^i} x^{-\sum_j b_j 2^j} = \prod_i g^{a_i 2^i} \prod_j x^{-b_j 2^j}.$$

Therefore we may compute the contents of the third register for the price of two modular exponentiations so that the above counts must be doubled once the machine model and scratch space has been designated. Let us say that we have decided on T scratch bits and a machine model yielding a count for modular exponentiation of $[\overline{a}]$ gates and P pulses. Then to this point we have used $3L + 2 + T$ scratch bits, $2P + 50L - 19$ pulses and a gate count of $[0, 4L-1, 4L-2] + 2[\overline{a}]$.

Finally we perform a Fourier transform A_q, on each of the first two registers to yield the state:

$$\frac{1}{(p-1)q} \sum_{a,b=0}^{p-2} \sum_{c,d=0}^{q-1} exp(\frac{2\pi i}{q}(ac+bd))|c, d, g^a x^{-b}(mod\ p)>.$$

Here q is the power of 2 such that $p < q < 2p$, i.e. $q = 2^L$. The operation A_q requires $(L+1)(2(L+1)-1)$ laser pulses so the final number of pulses is now $2P + 2L^2 + 53L - 18$.

The following chart lists the storage and pulse requirements assuming the enhanced machine model with $2L+2$ scratch qubits.

L	qubits	pulses
100	504	3.64×10^8
500	2504	4.3×10^{10}
1000	5004	3.71×10^{11}

For comparison, note that the estimates in [2] for factoring a 500 bit number require 2501 qubits of storage and about 3×10^8 laser pulses.

References

[1] Vedral, Barenco, and Ekert. *Quantum Networks for Elementary Arithmetic Operations.* Los Alamos Physics preprint quant-ph/9511018.

[2] Beckeman, Chari, Devabhaktuni, and Preskill. *Efficient Networks for Quantum Factoring.* Los Alamos Physics preprint quant-ph/9602016.