International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:1, No:10, 2007

A Post Processing Method for Quantum Prime Factorization Algorithm based on Randomized Approach

Mir Shahriar Emami, and Mohammad Reza Meybodi

***Abstract*—**Prime Factorization based on Quantum approach in two phases has been performed. The first phase has been achieved at Quantum computer and the second phase has been achieved at the classic computer (Post Processing). At the second phase the goal is to estimate the period r of equation $x^r \overset{N}{\equiv} 1$ and to find the prime factors of the composite integer N in classic computer. In this paper we present a method based on Randomized Approach for estimation the period r with a satisfactory probability and the composite integer N will be factorized therefore with the Randomized Approach even the gesture of the period is not exactly the real period at least we can find one of the prime factors of composite N. Finally we present some important points for designing an Emulator for Quantum Computer Simulation.

***Keywords*—**Quantum Prime Factorization, Randomized Algorithms, Quantum Computer Simulation, Quantum Computation.

I. Introduction

FACTORIZING large integers has been an important problem from past till now especially in the state of RSA technique [1, 2]. In the RSA data encryption technique it is needed to find a very large integer that equals to multiplication of two large prime numbers. Many types of Prime Factorization methods presented but none of them even the methods based on parallelism could factorize a composite integer at a polynomial time [3-8]. Quantum Approach powered by Quantum Parallelism, Entanglement [9] and etc can solving the algorithmic problems in polynomial time for instance Prime Factorization Quantum Algorithm solved by Peter Shor [10-13]. If the composite integer N is known the goal is solving (1). Assume x is a random variable such that gcd (N, x) =1:

$$x^r \overset{N}{\equiv} 1 \tag{1}$$

Assume r is even then we have:

$$x^r - 1 \overset{N}{\equiv} 0 \Rightarrow (x^{\frac{r}{2}} - 1)(x^{\frac{r}{2}} + 1) \overset{N}{\equiv} 0 \tag{2}$$

Mir Shahriar Emami is Member of the Board of Islamic Azad University, Roudehen Branch, Technical and Engineering Faculty, Computer Engineering Group, Roudehen, Iran.

Mohammad Reza Meybodi is the Professor of Computer Engineering and Information Technology Department, Amirkabir University of Technology, Tehran, Iran.

It means than at least one of the P1 or P2 in (3) is a prime factor of composite N, although some of the answers may be 1 or/and the N (the trivial factors):

$$\begin{cases} P1 = \gcd\left((x^{\frac{r}{2}} - 1) ModN\ , N\right) \\ P2 = \gcd\left((x^{\frac{r}{2}} + 1) ModN\ , N\right) \end{cases} \tag{3}$$

Prime Factorizing of composite N to be achieved through two phases:

A. First Phase

The simultaneously calculation of $x^t ModN$ such that $0 \le t \le N$ and then performing the Quantum Fourier Transformation at the Quantum Computer.

B. Second Phase

The period estimation that satisfies Equation 1 at the classic computer.

At first by the reason of the Quantum Computer is not accessible anywhere widely I've designed a Simulator for Quantum Computer including two Quantum Register such that the basic concepts of Quantum Computer for instance: Quantum bit or Qubit(This word was invented first time by Schumacher in 1993), Quantum Register, Entanglement, Quantum Measurement (a destructive phenomena),...[14-16] have been simulated and then the necessary Unitary Functions for example Quantum Furrier Transform[16-18] has been Simulated too. But the only difficulty is the Estimation of Period of (1) therefore we present a method based on Randomized Approach for estimating the period.

II. The Implementation of Quantum Algorithm

If the composite number N have been assumed we are going to find an integer q such that: $q=2^L$ and L ∈ ℕ then we must have:

$$N^2 < q < 2N^2 \tag{4}$$

At this computer two Entangled Quantum Registers named 1 and 2 has been used such that a and b are the binary vectors of Quantum Register 1 and 2.The system state at each time can be express as:

$$|\psi>=\sum_{a=0}^{q-1}\sum_{b=0}^{q-1}E_{a,b}\ |a,b> \tag{5}$$

In which $E_{a,b}\in$ *Complex* and we have:

$$\sum_{a,b}|E_{a,b}|^2=1 \tag{6}$$

Executing the Quantum Algorithm for Prime Factorizations has been performed in 6 steps as expressed bellow:

A. Step 1

We start the Quantum Computer with initial state $|\psi\rangle=|0,0\rangle$ and then we should use Hadamard Unitary Operation on Quantum Register 1 for achieving superposition of all states 0 to q-1 in this register, now the state of the system can be expressed as:

$$|\psi>=\frac{1}{\sqrt{q}}\sum_{a=0}^{q-1}|a>|0> \tag{7}$$

B. Step 2

At this step we are going to guess a random value and storing it in the random variable x such that gcd (x , N) = 1 and now we using function $F_a= x^a$ Mod N on Quantum Register 1 and storing the result on Quantum Register 2 .The state of the system can be expressed as:

$$|\psi\rangle=\frac{1}{\sqrt{q}}\sum_{a=0}^{q-1}|a>|0>\xrightarrow{F_a|\Psi\rangle}\frac{1}{\sqrt{q}}\sum_{a=0}^{q-1}|a>|x^a Mod\ N> \tag{8}$$

C. Step 3

Now we measure the value of Quantum Register 2.As we know the measurement of a Quantum Register is a destructive action therefore we must collapse the values of Quantum Register 2 .Assume the measurement value is K .As the Quantum Registers 1 and 2 are entangled so the values in Quantum Register 1 will be collapsed just after we measure the Quantum Registers 2 and in the Quantum Registers 1 there will be only values that satisfies (9) it means:

$$\ni a: x^a ModN = K \tag{9}$$

So we get a set named |A| includes the values of 0 to q-1 which satisfies (9); we name the element of this set a' and the quantity of a's is U, it means:

$$\|A\|=U \qquad \text{(Read Norm A)} \tag{10}$$

So the state of the system is:

$$|\psi>=\frac{1}{\sqrt{\|A\|}}\sum_{a'\in A}|a',K> \tag{11}$$

It is clear that the set A is as (12) which $U\approx\frac{q}{r}>>1$ it means:

$$A=\{a_0,a_0+r,a_0+2r,...,a_0+(U-1)r\} \tag{12}$$

So we can write the state of the system as bellow:

$$|\psi>=\frac{1}{\sqrt{U}}\sum_{d=0}^{U-1}|\overbrace{a_0+d_r}^{a'},K> \tag{13}$$

D. Step 4

Now we perform Quantum Furrier Transform on Quantum Register 1 then we get:

$$\frac{1}{\sqrt{U}}\sum_{d=0}^{U-1}|a_0+dr,K>\xrightarrow{QFT}$$

$$\frac{1}{\sqrt{qU}}\sum_{c=0}^{q-1}\sum_{d=0}^{U-1}e^{\left(2\pi\frac{E(a_0+dr)}{q}\right)}|E,K>$$

$$=\sum_{c=0}^{q-1}\frac{e^{\frac{2\pi iEa_0}{q}}}{\sqrt{qU}}\sum_{d=0}^{U-1}e^{\left(2\pi\frac{Edr}{q}\right)}|E,K>$$

And then we get the state of the system like we show at the (14) which $\rho=e^{\frac{2\pi iEr}{q}}$:

$$|\psi\rangle=\sum_{c=0}^{q-1}\frac{e^{\frac{2\pi iEa_0}{q}}}{\sqrt{qU}}\left(\sum_{d=0}^{U-1}\rho^d\right)|E,K> \tag{14}$$

E. Step 5:

If we measure the Quantum Register 1 the probability of observation of state |E> is equal:

$$P(|E\rangle)=\frac{1}{qU}\left|\sum_{d=0}^{U-1}\rho^d\right|^2 \tag{15}$$

And with considering the ρ as $e^{\frac{2\pi iEr}{q}}$, we can rewrite the Equation 15 as (16):

$$P(|E\rangle)=\frac{1}{qU}\left|\sum_{d=0}^{U-1}\left(e^{2\pi\frac{Er}{q}}\right)\right|^2 \tag{16}$$

Now we discuss on $\frac{Er}{q}$ ratio. Two conditions may be occurred on this ratio as expressed bellow:

1)CONDITION 1

If the value of the $\frac{Er}{q}$ ratio be very small it means:

$$\frac{Er}{q} << 1 \tag{17}$$

Whereas: $\sum_{d=0}^{U-1}\rho^{i} = \frac{1-\rho^{U}}{1-\rho} \approx \frac{1-1}{1-1} = 0$

So the probability of observing the state |E> is:

$$P(|E\rangle) \approx \frac{1}{qU}(0) \Rightarrow P(|E>) \approx 0 \tag{18}$$

It means this probability is approximately zero!

2)CONDITION 2

If the $\frac{Er}{q}$ ratio almost be as large as a positive integer like s such that we have $\frac{Er}{q} \approx s$, therefore we have:

$$\rho = e^{2\pi i * s} \tag{19}$$

Whereas based on Euler Equation we can write ρ as:

$$\rho = \cos(2\pi s) + i\sin(2\pi s) \Rightarrow \rho = 1$$

So the probability of observing the state |E> is:

$$P(|E\rangle) = \frac{1}{qU}\left|\sum_{d=0}^{U-1}1\right|^{2} = \frac{1}{qU}|U|^{2} \Rightarrow$$

$$P(|E\rangle) = \frac{U}{q} \tag{20}$$

Therefore the probability of observing the state |E> is large so we can almost predict (21) Such that the r and s variables are Natural numbers:

$$\frac{E}{q} \cong \frac{s}{r} \tag{21}$$

F. Step 6:

Finally after observing the state |E> and with known q the $\frac{E}{q}$ ratio is known. Now the problem is how we can guess practically the period r with classical ways at the classic computer? So at first we consider to use Continued Fractions Method base on Randomized Approach, it means we are going to find the ratios bellow such that the limit of them goes to $\frac{s}{r}$ so:

$$\frac{s_1}{r'_1} \approx \frac{s_2}{r'_2} \approx \ldots \frac{s_j}{r'_j} \ldots \cong \frac{s}{r} \tag{22}$$

Now through an iteration action we consider to evaluate which of the r'j s satisfies the (1). Each of r'j s satisfies (1) that is the period we are going to find! It is better to perform the LCM of results because of finding the smallest period. After finding the period r with a high priority at least one of the prime factors of composite N with the (3) has been got.

In fact when the unknown |E> state has been measured by my own Emulator which I designed, I immediately was calculating the $\frac{E}{q}$ ratio with Continued Fractions Method and was estimating r'_j ,the probabilistic period, and I immediately was checking if r'_j satisfied (1) whereas many of the results didn't satisfy (1) I had to drop out my results and continued with the Continued Fractions Method more times. Occasionally I had to repeat this way till the answer had been greater or equal to N and checked the satisfaction of the new result in (1). If r'_j would be odd I doubled r'_j then checked the satisfaction of (1) and if all the results was failed I retry with another guessing of random variable x. Therefore the post processing phase that concern with the estimation of period had been taken a long time. For this reason I applied Randomized Approach for guessing the period. My opinion was that, after measuring the state |E> without attention to (1), I supposed that measured value probably would get the period! I continued with (3) for trying to find at least one of the prime factors. Practically the result was satisfactory and at most conditions, one of the prime factors of composite N was got! Interestingly the probably periods often didn't satisfy (1) (Look Appendix)! Consequently if the the estimation of period r is exactly be the real period with a high probability we can both prime factors of composite N! But if our period gesture is not exactly the real period with a high probability at least we can find one of the prime factors of composite N therefore we can find another prime factor easily!

III. QUANTUM COMPUTER SIMULATION IMPORTANT POINTS

As for executing and evaluating a Quantum Algorithm for Prime Factorization I need a Quantum computer because I knew this computer didn't accessible widely therefore I've designed an Emulator for Quantum computer. In this Emulator I've considered many points as bellow:

A. As the Quantum computers are based on Quantum mechanics and the Quantum mechanics base on Complex Probabilistic Theory so at first I designed a class for complex numbers. At this class I've defined two main properties includes: *Imaginary* and *Real* that concerns the two parts of a complex number. For working with the complex variables many methods and operators was needed. Operators + , -and * were the main operators that had been needed for performing

calculations on complex variables and the point is that we must have high calculating accuracy.

B. In a classic computer the smallest unit for saving data is a bit but at Quantum physics the smallest unit for saving data is a qubit then I have to simulate qubit by a class named Qubit. Qubit variable includes two basic states |0> (zero state) and |1> (one state) which with a probability like α^2 occurs |0> and another probability like β^2 occurs |1> such that: $\alpha^2 + \beta^2=1$. For working with Qubit variables I defined some methods, the main of them was the methods for putting and getting probability value of a qubit therefore I've made SetVal and GetVal methods.

C. In fact for storing a data we need a larger space than a qubit therefore we need some quantum registers. A quantum register with size n can store 2^n number simultaneously so we need a large space of memory in the classic computer, in this reason I made a class named QuRg. In QuRg variable we must have two main properties including: qrSize for storing the size of a quantum register and qrState for storing the all possible values in Superposition condition. One of the important methods on this class is the Measure method. The Measure method must be destructive, it means that after measuring a quantum register, it has to collapse and the quantum register must loose all of its values only the values which measured. Another important point is the summation of probability of states in a quantum register because there is a limitation. For a quantum register with size 2 like as $|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$ the limitation is $\sum_{i=0}^{3} |\alpha_i|^2 |i\rangle = 1$ (It is said Normalized) therefore we need a method for normalizing the value of quantum registers after performing a Unitary operation.

D.Almost all the quantum algorithms concern to Unitary functions therefore we need a class for this reason named UniFunc.In the Prime Factorization algorithm the most important Unitary Function is Quantum Furrier Transform for finding the period so I've designed a Unitary function named QFT and for this reason I used the two equations bellow :

$$\begin{cases} real = \frac{1}{\sqrt{q}} Cos\left(2\pi(a\frac{E}{q})\right) \\ imaginary = \frac{1}{\sqrt{q}} Sin\left(2\pi(a\frac{E}{q})\right) \end{cases} \quad (23)$$

IV. Conclusion

1) The second phase of executing quantum algorithm for prime factorizing is the Estimation of Period, for this reason with a high probability we can estimate the period based on Randomized Approach rapidly.
2) With Randomized Approach even the gesture of the period is not exactly the real period at least we can find one of the prime factors of composite N.
3) Randomized Approach with a satisfactory probability for solving many of the problems can get at least some of the answers.
4) As we want to execute and evaluate a Quantum algorithm whereas a Quantum computer isn't accessible widely, we need an Emulator for Quantum computer.
5) For designing a Quantum computer Emulator we have to consider many points that concern to Quantum physics and Quantum mechanics like as Complex Probability Theory, Entanglement, Quantum Register measurement, Side effects of Unitary operations, Superposition, …till we can get best results.

References

[1] R. L. Rivets and R. D. Silverman, Are Strong Primes Needed for RSA? , 22-Nov-1999
http://citeseer.ist.psu.edu/rivest99are.html
[2] R. Crandall and C. Pomerance, Prime Numbers: A Computational Perspective, Springer, ISBN: 0387252827, 2005
[3] H. Riesel, Prime Numbers and Computer Methods for Factorization, Birkhauser, Quinn-Woodbine, USA, ISBN:0-8176-3743-5
[4] J. Franke and T. Kleinjung, C. Paar, J. Pelzl, C. Priplata, C. Stahlke, M. Šimka, An Efficient Hardware Architecture for Factoring Integers with the Elliptic Curve Method, University of Bonn, University of Bochum, EDIZONE GmbH, Bonn, University of Košice, 24-2-2005
http://cr.yp.to/bib/2005/franke-ecm.pdf
[5] O. Åsbrink, J. Brynielsson, Factoring large integers using parallel Quadratic Sieve,Apr- 2000
www.nada.kth.se/~joel/qs.pdf
[6] R. P. Brent, Recent Progress and Prospects for Integer, Oxford University, Oxford, UK
http://www.comlab.ox.ac.uk
[7] H. T. Riele, Factoring Large Numbers: Fun or Applied Science? , Research project: Computational Number Theory and Data Security, CWI-AR 1999
http://dbs.cwi.nl:8080/cwwwi/owa
[8] R. P. Brent, Parallel Algorithms for Integer Factorization, Australian National University, Canberra, Canada
Fttp://ftp.comlab.ox.ac.uk/pub/Documents/techpapers/Richard.Brent/rpb115.ps.gz
[9] S. L. Braunstein, Quantum computation, Computer Science, University of York, UK,23-Aug 1995
www.weizmann.ac.il/chemphys/schmuel/comp/comp.html
[10] V. Vedral, Introduction to Quantum Information Science, Part III, Section 11,12,13, Oxford University Press, Oxford, UK, 2006
[11] M. Le Bellac, A Short Introduction to Quantum Information and Quantum Computation, Cambridge University Press, Cambridge, UK, 2006
[12] L. Ip, Shor's Algorithm is Optimal, University of California, Berkeley, USA, 5,-Nov-2003
http://citeseer.ist.psu.edu/631220.html
[13] S. J. Lomonaco and S. J. Kauffman, A Contiuous Variable Shor Algorithm, arXiv: quant-ph/0210141 v2, 8-Jun-2004
[14] L. Fortnow, Introduction of Quantum Computing from the Computer Science Perspective and Reviewing Activities, Nec Res. And Develop, Vol 44, No3, Jul-2003.
[15] A. M. Steane and E. G. Rieffel, Beyond Bits: The Future of Quantum Information Processing, Page 38, IEEE, Jan-2000
[16] R. T. Perry, The Temple of Quantum Computing, online e-book, Dec-19,-2004
www.phys.ntu.edu.tw/goan/Courses/D3040/PHYS_D3040.html
[17] C. Moore, D. Rockmore and A. Russell, Generic quantum Fourier transforms
ACM Transactionson Algorithms (TALG), archive Volume 2 , Issue 4, Pages: 707 - 723 , Oct-2006 , ISSN:1549-6325
http://portal.acm.org/citation.cfm
[18] J. J. Vartiainen, Unitary Transformation for Quantum Computing, Department of Engineering Physics and Mathematics, Helsinki University of Technology, Apr-2005
http://lib.tkk.fi/Diss/2005/isbn9512276127

Appendix

N	x	q	E	s_j/r'_j	r'_j	r	x^r Mod N	$(x^{r/2}-1)$ Mod N	$(x^{r/2}+1)$ Mod N	P1	P2
21	2					6	1	7	9	7	3
21	2	512	68	17/128	128	128	16	15	17	3	1
21	5	512	7	1/72	73	146	4	3	5	3	1
21	11	512	7	1/73	73	146	16	10	12	1	3
33	3	2048	1638	819/1024	1024	1024	15	24	26	3	1
33	4	2048	528	33/128	128	128	31	24	26	3	1
33	4	2048	1059	469/907	907	1814	25	15	17	3	1
33	5	2048	410	205/1024	1024	1024	31	24	26	3	1
33	7					10	1	9	11	3	11
33	17	2048	288	9/64	64	64	31	24	26	3	1
33	17	2048	374	187/1024	1024	1024	31	24	26	3	1
35	2	2048	238	119/1024	1024	1024	16	10	12	5	1
35	3	2048	120	15/256	256	256	11	15	17	5	1
35	4	2048	16	1/128	128	128	16	10	12	5	1
35	9	2048	309	43/285	285	570	1	28	30	7	5
35	11	2048	154	77/1024	1024	1024	11	15	17	5	1
35	13	2048	175	37/433	433	866	29	12	14	1	7
39	2					12	1	24	26	3	13
39	2	2048	264	33/256	256	256	16	21	23	3	1
39	5	2048	111	20/369	369	738	25	4	6	1	3
39	11	2048	324	81/512	512	512	22	15	17	3	1
39	17	2048	88	11/256	256	256	22	15	17	3	1
39	23	2048	334	167/1024	1024	1024	16	21	23	3	1
51	2	4096	512	1/8	8	8	1	15	17	3	17
51	7	4096	256	1/8	8	16	1	15	17	3	17
51	11	4096	256	1/16	16	16	1	15	17	3	17
51	19	4096	1007	311/1265	1265	2530	4	18	20	3	1
55	2					20	1	33	35	11	5
55	2	4096	246	123/2048	2048	2048	36	15	17	5	1
55	3	4096	819	1/5	5	10	34	22	24	11	1
55	7	4096	508	127/1024	1024	1024	36	15	17	5	1
55	19	4096	78	39/2048	2048	2048	16	25	27	5	1
65	2					12	1	63	0	1	65
65	2	8192	1330	665/4096	4096	4096	16	60	62	5	1
65	3	8192	874	437/4096	4096	4096	16		60	65	5
65	7	8192	1630	815/4096	4096	4096	61		15	17	5
65	11	8192	683	171/2051	2051	4102	36		5	7	5
65	13	8192	1366	683/4096	4096	4096	61		15	17	5