Quantum Computing for Solving a System of Nonlinear Equations over GF(q)

Essam Al Daoud
Computer Science Department, Zarqa Private University, Jordan

Abstract: *Grover's quantum search algorithm is one of the most widely studied and has produced results in some search applications faster than their classical counterpart by a square-root. This paper modifies Grover's algorithm to solve nonlinear equations over Galois Finite field GF(q) in $O(\sqrt{2^{mn}})$ iteration, while the best classical general solution takes $O(2^{nm})$ iteration. The modification is done by using a register for each variable and represent it by n qubits. The paper also introduces the implementation of the suggested algorithm by using the simulator QCL 5.1.*

Keywords: *Quantum computing, quantum operations, nonlinear equations, quantum simulator.*

Received September 19, 2005; accepted May 14, 2006

1. Introduction

Quantum computation and quantum information are new fields in the computer science which rapidly gaining popularity and earning a lot of attention in the last decade. The main advantage of the quantum technology is the possibility to solve the hard problems efficiently, such as integer factorization, finding the hidden subgroup, lattice problems, and in general solving the NP complete problems in polynomial time. Furthermore quantum information offers a new cryptosystem suitable for the modern communication and computation.

In 1985, David Deutsch developed the quantum Turing machine, showing that quantum circuits are universal and can simulate any other Turing machine efficiently [9]. In 1994, Shor showed how to factorize very large integers into primes, using a quantum algorithm that is exponentially faster than the best classical factoring algorithm [12]. The key idea of quantum factoring algorithm is the use of a Fourier transform to find the period of a sequence. Shor's algorithm could theoretically break many of the cryptosystems in use today such as RSA and elliptic curve cryptography. In 1996, Lov Grover developed an algorithm to perform quantum search, which was quadratically faster than a classical computing search, Grover's algorithm can identify an item from an unsorted list with N entries in $O\sqrt{N}$ steps and using *O (*log*N)* storage space [4, 5], it can also be used for solving the collision problem, breaking Data Encryption Standard (DES), and estimating the median of a set of numbers [3, 7]. In 2001, IBM's Almaden Research Center demonstrated the execution of Shor's algorithm using 7-qubit NMR computer. The number 15 was factored using identical molecules, each containing 7 atoms [6].

The remainder of this paper is organized as follows. Section 2 presents the basic concepts of the quantum computer, and the superposition representation of *n* qubits. Section 3 derives the quantum operations or gates from Schrödinger equation. Section 4 introduces the principles of the quantum algorithms such as quantum parallelism, interference and the measurement. In section 5, we introduce a new quantum algorithm to solve a system of nonlinear equations over GF(q). In section 6, we implement the suggested quantum algorithm by using the simulator QCL 5.1. Finally, section 7 concludes the paper.

2. The Basic Concepts of the Quantum Computer

The basic unity information in the quantum computer is the qubit, which has two possible states |0> or |1>, this can be realized by the spin of a particle, the polarization of a photon or by the ground state and an excited state of an ion. Unlike classical bits, a qubit can be forced into a *superposition* of the two states which is often represented as linear combination of states:

$$|\psi> = \alpha\ |0> + \beta\ |1>$$

for some α and β such that $|\alpha|^2 + |\beta|^2 = 1$. There is no good classical explanation of superpositions: A quantum bit representing 0 and 1 can neither be viewed as between 0 and 1 nor can it be viewed as a hidden unknown state that represents either 0 or 1 with a certain probability. However; the processes in the quantum computer are governed by Schrödinger equation which has no classical explanation.

The quantum states can be represented as vectors in Hilbert space rather than classical variables such that:

$$|0>=\begin{pmatrix}1\\0\end{pmatrix} \text{ and } |1>=\begin{pmatrix}0\\1\end{pmatrix}$$

and the superposition state is:

$$|\psi>=\alpha\begin{pmatrix}1\\0\end{pmatrix}+\beta\begin{pmatrix}0\\1\end{pmatrix}=\begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

The state of *n* qubits (a register) is represented by the tensor (⊗) product of the individual states of the qubits in it. For example, if we have two qubits in a register, and both have the state $|0\rangle$ then the register status is $|00\rangle$, which corresponds to the vector

$$|0>\otimes|0>=|00>=\begin{pmatrix}1\\0\end{pmatrix}\otimes\begin{pmatrix}1\\0\end{pmatrix}=\begin{pmatrix}1\\0\\0\\0\end{pmatrix}$$

Similarly, $|01\rangle$, $|10\rangle$, and $|11\rangle$ correspond to the vectors:

$$\begin{pmatrix}0\\1\\0\\0\end{pmatrix},\begin{pmatrix}0\\0\\1\\0\end{pmatrix}, \text{ and } \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

Respectively. The superposition of a 2-qubit register is:

$$|\psi>=\begin{pmatrix}\alpha_1\\\beta_1\end{pmatrix}\otimes\begin{pmatrix}\alpha_2\\\beta_2\end{pmatrix}=\begin{pmatrix}\alpha_1\alpha_2\\\alpha_1\beta_2\\\beta_1\alpha_2\\\beta_1\beta_2\end{pmatrix}$$

Where, $|\alpha_1\alpha_2|^2 + |\alpha_1\beta_2|^2 + |\beta_1\alpha_2|^2 + |\beta_1\beta_2|^2 = 1$. This can be generalized to *n* qubits easily [9].

3. Quantum Operations

The second postulate of quantum mechanics describes the evolution of a closed system by the Schrödinger equation:

$$ih\frac{\partial}{\partial t}|\psi>=H|\psi>$$

Where H is the Hamiltonian operator and h is Planck's constant. In quantum physics, it is common to use a system of measurement where h = 1, so the discrete-time solution of Schrödinger equation is:

$$|\psi>=U|\psi_0>$$

Where U is a unitary matrix. A general 2-dimensional complex unitary matrix U can be written as:

$$U=e^{itH}$$

The common single qubit unitary operations or gates on registers contain a qubit [10, 11]:

Pauli Gates:

$$X=\begin{pmatrix}0&1\\1&0\end{pmatrix}, Y=\begin{pmatrix}0&-i\\i&0\end{pmatrix}, Z=\begin{pmatrix}1&0\\0&-1\end{pmatrix}$$

Hadamard Gate:

$$H=\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}$$

Phase- and π/8-Gate:

$$S=\begin{pmatrix}1&0\\0&i\end{pmatrix}, T=\begin{pmatrix}1&0\\0&e^{i\pi/4}\end{pmatrix}$$

The previous gates can be generalized to registers thatcontain n qubit by applying tensor (⊗) product n times on the unitary operation itself. on the other hand, the two common qubit operations or gates:

Controlled-Not Gate:

$$CNot=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix} \quad or \quad CNot\,|x,y>\longrightarrow|x\oplus y,y>$$

Swap-Gate:

$$Swap=\begin{pmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{pmatrix} \quad or \quad Swap\,|x,y>\longrightarrow|y,x>$$

However, the universal set of quantum gates can be built by using CNot and single qubit operations which can be implemented by using a beam splitter and applying a radio frequency pulse.

4. Quantum Algorithm Principles

The superposition of n qubits (or a register) allows each operation or quantum gate acts on all basis states simultaneously, this type of computation is the basis for quantum parallelism which leads to a completely new model of data processing. Shor's algorithm is a good example of quantum superposition and parallelism. Let $|\psi>=|0>|0>$ be the initial state of a quantum computer, then the Hadamard operation on the first register leaves the quantum computer in the following superposition state:

$$|\psi>=\frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}|i>|0>$$

Quantum parallelism exploited by applying a reversible function f on all states from $|0>$ to $|2^n-1>$

simultaneously. In Shor's algorithm, $f(x) = x^i \mod n$, and the computer state becomes:

$$|\psi>=\frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}|i>|x^i \bmod n>$$

However, the observation of the superposition of states makes it collapse to one of the states with a certain probability. For example, if we like to measure the quantum register:

$$\psi>=\frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}|i>$$

Then the superposition states will collapse to the state |x> with probability:

$$p(x)=<\psi|M_x^t M_x|\psi>$$

and the state of the register after measurement:

$$|\psi^{'}>=\frac{M_x|\psi>}{<\psi|M_x^t M_x|\psi>}$$

Where $M_x = |x><x|$. Fortunately, quantum interference can be used to improve the probability of obtaining a desired result by constructive interference and minimize the probability of obtaining an unwanted result by destructive interference. Thus, the challenge is to design quantum algorithms which utilize the interaction of the superposition states to maximize the chance of the interesting states [10, 11].

5. Quantum Algorithm for Solving a System of NLE Over GF(q)

Solving a system of nonlinear equations over Galois Finite field GF(q) (NLE) is an NP hard problem and the best known general solution is brute force. One of the most important applications of NLE is the cryptanalysis, where many cryptography methods can be expressed as a system of quadratic equations over GF(q).

A system of nonlinear equations with m variables can be written as follows:

$$f_j(i_1,i_2,...,i_m)=0\text{, for all } j = 1, 2, ..., k$$

and the variables $i_1, i_2, ..., i_m$ are in the finite field *GF(q)*, where q is a prime number of length n or $q = 2^n$. The best solution of above systems on the classical computers takes O (2^{nm}) iteration, while this complexity can be reduced to O($\sqrt{2^{mn}}$) iteration on the quantum computers.

The following quantum algorithm can be used to solve nonlinear equations over GF(q) in O ($\sqrt{2^{mn}}$) iteration:

1. Initialize m register to the state | 0>, where m is the number of variables.
2. Use n qubits in each register, where n is the number of bits in q, and Let $z = 2^{nm}$.
3. Convert the registers to the superposition states, thus the system status:

$$|\psi> = \frac{1}{\sqrt{z}}\sum_{i_1=0}^{2^n-1}\sum_{i_2=0}^{2^n-1}...\sum_{i_m=0}^{2^n-1}|i_1>|i_2>...|i_m>$$

4. Let s = $\lfloor(\pi*\sqrt{z})/4\rfloor$.
5. Repeat the steps 6-9 *s* times.
6. Change the state $|i_1>|i_2>...|i_m>$ to $-|i_1>|i_2>...|i_m>$ if and only if $f_j(i_1,i_2,...,i_m)=0$ for all *j=1, 2,..., k*.
7. $|\psi>=\overset{nm}{\bigotimes}H\;|\psi>$.
8. Change the state $|i_1>|i_2>...|i_m>$ to $-|i_1>|i_2>...|i_m>$ if and only if i_j = *0* for all *j=1, 2,...,m*.
9. $|\psi>=\overset{nm}{\bigotimes}H\;|\psi>$.
10. Observe thes ystem.

For simplicity, we will assume that the system of nonlinear equations has an unique solution $x_1,x_2,...,x_m$. Let $|A>=|x_1>|x_2>...|x_m>$, where $x_1,x_2,...,x_m$ are the correct solution and $|B>$ = $\sum_{i_1=0}^{2^n-1}\sum_{i_2=0}^{2^n-1}...\sum_{i_m=0}^{2^n-1}|i_1>|i_2>...|i_m>$, *For all* $x_j \neq i_j$, *j = 1, 2.., m*, then the system status at any iteration $|\psi>$ = *r* |*A*> + *t* | *B*>, but after performing the steps 6-9 the system status is:

$$|\psi> = (\frac{z-2}{z}r+\frac{2(z-1)}{z}t)|A>+(\frac{-2}{z}r+\frac{z-2}{z}t)|B>$$

Let $\sin^2\beta=1/z$, then we can show that by induction the system status after *l* iterations becomes:

$$|\psi>=\sin((2l+1)\beta)|A>+\frac{\cos((2l+1)\beta)}{\sqrt{z-1}}|B>$$

The superposition states $|\psi>$ will collapse to the correct state |A> with high probability if and only if *sin ((2l + 1)*β is close to one. Thus the number of the required iterations is:

$$l=\left\lfloor\frac{\pi}{4\sin^{-1}(1/\sqrt{z})}\right\rfloor\approx\left\lfloor\frac{\pi}{4}\sqrt{z}\right\rfloor$$

However, we can obtain the correct solution with zero failure rate by using the modified version of Grover's algorithm which introduced by Long [8].

6. Quantum Algorithm Simulation

Although the quantum computers at this point in time are not efficient, many quantum simulators have been developed such as: OpenQubit, QCL and QuantumOctave [1, 10, 11]. Quantum Computation Language (QCL) is a high level, architecture independent programming language for quantum computers that includes program files for simulation of an implementation of Shor's algorithm and files for simulating other aspects of quantum computation. Figure 1 shows the implementation of the suggested quantum algorithm by using QCL 5.1, the implementation is restricted to 3 variables and any field size, but it can be generalized to any number of variables.

```
// m: The number of  the registers, but here is restricted to
//3 registers
// n: The number of the qubits in the registers


Procedure find (int m, int n)
 {
   Int s = floor (pi * sqrt (2 ^ (m * n)) / 4);  // the number of the
                                                   //iterations
   Int i;
   Qureg x1 [n]; qureg x2 [n]; qureg x3 [n];
     reset;
   Mix (x1 & x2 & x3);   // convert the registers to the
                          // superpositions
    For i = 1 to s
      {
         Mark (x1, x2, x3);       // call  $f_j(i_1, i_2, i_3) = 0$
                                  //and mark the correct solution
         Mix(x1&x2&x3);           // apply $\overset{nm}{\otimes} H$
         Not(x1&x2&x3);
         // mark  x  if and only if $i_j = 0$  for all j=1, 2, ..., m.
         If x1&x2&x3
           {
              Phase(pi);
           }
         Not (x1& x2 & x3);
         Mix(x1& x2 & x3);          //  apply $\overset{nm}{\otimes} H$
         If  x1[0]  { Phase(pi); }
         If not x1[0]
          {
             Phase (pi);
          }
         Measure  x1& x2 & x3          //Observe the System
      }
}
```

Figure 1. The implementation of the suggested quantum algorithm by using QCL 5.1.

7. Conclusion

Quantum computing outperforms the best classical techniques for some hard computation problems. The universal set of quantum gates can be built by using CNot and single qubit operations which can be implemented by using a beam splitter and applying a radio frequency pulse. This paper introduces a quantum algorithm to solve a system of nonlinear equations over Galois Finite field GF (q) in O ($\sqrt{2^{mn}}$) iteration, which is considered faster than their classical counterpart by a square-root. The new algorithm uses a register for each variable and represents it by n qubits. The suggested algorithm can be implemented by using the simulator QCL 5.1.

References

[1] Blaha S., "Quantum Computers and Quantum Computer Languages: Quantum Assembly Language and Quantum C Language," Cornell University Library, available at: http://arxiv.org /abs/quant-ph/0201082, 2002.

[2] Brassard G., "New Trends in Quantum Computation," *in Proceedings of the 13th Annual Symposium on Theoretical Aspects of Computer Scienc*e, Grenoble, France, Springer, pp. 3-10, 1996.

[3] Brassard G., Hoyer P., and Tapp A., "Quantum Counting," *in Proceedings of 25th International Colloquium on Automata, Languages, and Programming (ICALP98)*, Berlin, pp. 820-831, 1998.

[4] Grover L. K., "A Fast Quantum Mechanical Algorithm for Database Search," *in Proceedings of the 28th Annual ACM Symposium on the Theory of Computing*, Philadelphia, PA, pp. 212-219, May 1996.

[5] Grover L. K., "Quantum Search on Structured Problems," Cornell University Library, available at: http://arxiv.org/abs/quant-ph/9802035, pp. 1695-1705, 1999.

[6] IBM, www.ibm.com, available at: http://domino. research.ibm.com/comm/bios.nsf/pages/quantum. html, March 2005.

[7] Kuperberg G., "A Subexponential-Time Quantum Algorithm for the Dihedral Hidden Subgroup Problem," Cornell University Library, available at: http://arxiv.org/pdf/quant-ph/0302 112, 2004.

[8] Long G. L., "Grover Algorithm with Zero Theoretical Failure Rate," Cornell University Library, available at: http:// arxiv.org/abs/quant-ph/0106071, 2001.

[9] Nielsen M. A. and Chuang I. L., *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, 2000.

[10] Omer B., "Quantum Programming in QCL," *Master Thesis*, Technical University of Vienna, 2000.

[11] Omer B., "Procedural Quantum Programming," *in Proceedings of the 5th International Conference CASYS 2001*, Belgium, pp. 276-285 2002.

[12] Shor P. W., "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on

a Quantum Computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484-1509, 1997.

Essam Al Daoud received his BSc from Mu'tah University, MSc from Al Al-Bayt University, and his PhD in computer science from University Putra Malaysia in 2002. Currently, he is an assistant professor in the Computer Science Department at Zarqa Private University, Jordan. His research interests include quantum computing, cryptography, singular value decomposition and artificial intelligent.