



Simulación clásica de computación cuántica para el cálculo del valor propio de un operador hermítico

Adriana De Mendoza y John H. Reina

Grupo de Tecnología, Información y Complejidad Cuántica, Departamento de Física, Universidad del Valle, A.A. 25360, Cali, Colombia

Recibido 22 de Oct. 2007; Aceptado 2 de Sep. 2009; Publicado en línea 30 de Oct. 2009

Resumen

Clásicamente, el tiempo de cálculo de la energía de sistemas cuánticos crece exponencialmente con su tamaño. Sin embargo, este cálculo puede hacerse de orden polinomial usando un algoritmo cuántico recursivo tal como el denominado *algoritmo de estimación de fase* (PEA). El PEA reduce el número de bits cuánticos (qubits) requeridos para el registro de salida de veinte a cuatro aproximadamente. La función de onda del sistema molecular es mapeado a la base de los qubits de tal modo que el número de qubits requeridos escala linealmente con el número de elementos de la base y el número de compuertas necesitadas crece polinomialmente con el número de qubits. Este trabajo muestra cómo dicho algoritmo obtiene el valor propio de un operador hermítico, típicamente la energía del estado fundamental de un sistema físico abierto, dada la posible interacción con fuentes de ruido cuántico. Lo anterior se realiza mediante la modelación de fuentes de disipación cuántica dentro del algoritmo y la implementación de corrección cuántica de errores a través del algoritmo de Shor [1,2].

Palabras Clave: Algoritmos cuánticos, estimación de fase, corrección cuántica de errores

Abstract

Classically, the computational time required to calculate the energy of quantum systems scales exponentially with size, however, this calculation can be made in polynomial time using a recursive quantum algorithm termed *phase estimation algorithm* (PEA). This reduces the number of qubits required to read out a register from twenty to four approximately. The wave function for the molecular system is mapped out into the qubits basis such that the required number of qubits grows linearly with the number of basis functions, and the number of gates needed scales polynomially with the number of qubits. This work shows how the algorithm calculates the eigenvalue of an Hermitic operator, typically the ground state energy of an open quantum system, given the interaction with quantum noise sources. This is developed by modeling the sources of dissipation within the algorithm and realizing quantum error correction by means of Shor's algorithm [1,2].

Keywords: Quantum algorithms, phase estimation, quantum error correction

©2009. Revista Colombiana de Física. Todos los derechos reservados.

1. Introducción

Estamos familiarizados con la frustración de necesitar más recursos computacionales para resolver problemas de cómputo. Gran cantidad de problemas interesantes e importantes no son viables de solucionar en un computador convencional (“clásico”) debido a los desmesurados requerimientos de recursos computacionales.

Así, la promesa de la computación e información cuántica consiste en hacer posibles nuevos algoritmos que sean capaces de resolver ese tipo de problemas [3]. Un algoritmo cuántico consiste en un circuito de compuertas lógicas que alteran un bit cuántico de entrada (o un conjunto ellos) para producir un bit cuántico o un conjunto de bits de salida como resultado de la operación de cómputo. Cada compuerta cuántica es

una transformación unitaria que actúa sobre un número específico de qubits y es importante que el espacio de Hilbert del algoritmo sea descompuesto en el producto tensorial de los espacios bidimensionales de los qubits, asumiendo una descomposición natural en subsistemas, la cual es respetada por el sistema de compuertas de tal modo que estas actúan sobre pocos subsistemas a la vez.

El cálculo del valor propio de un operador Hermítico (la simulación completa de la ecuación de Schrödinger dependiente del tiempo), con algoritmos y computadores clásicos, crece exponencialmente con el tamaño del sistema, limitando así la complejidad de los sistemas y los potenciales a calcular, además de sacrificar la precisión que puede ser obtenida en dichos cálculos. La implementación de un algoritmo cuántico puede llevar a que ese tiempo de cálculo sea de orden polinomial. En el presente trabajo estudiamos un algoritmo cuántico recursivo que cumple dicho propósito, denominado *algoritmo de estimación de fase*: PEA [5,6] y presentaremos la implementación de la simulación clásica del mismo. Del mismo modo como la computación clásica es afectada por factores de ruido que pueden distorsionar o cambiar los resultados de cómputo, la computación cuántica es afectada por fuentes de ruido cuántico que alteran la información de las fases relativas obtenidas en el proceso de cálculo. Se introduce ruido al cálculo tomando el qubit de salida al final de cada etapa del algoritmo PEA y sometándolo al efecto de ruido cuántico mediante la operación de una compuerta unitaria U que introduce el error. Para proteger la información contra los efectos de ruido cuántico, el qubit es codificado mediante el algoritmo de Shor [1,2], el cual provee corrección para flípeo de bit, flípeo de fase, y combinaciones de estos[3].

2. Algoritmo de estimación de fase

Inicialmente se propone un estado fundamental $|\Psi\rangle$ (Función de onda de Hartree Fock) y se representa en un registro binario de qubits S tal que la función de onda del sistema de varias partículas se expresa en términos del estado de ocupación fermiónica de los orbitales atómicos. El número de orbitales es proporcional al número de átomos en la molécula. A partir del Hamiltoniano H del sistema se genera la transformación unitaria U , tal que

$$U|\psi\rangle = e^{iH\tau}|\psi\rangle = e^{i2\pi\phi}|\psi'\rangle, \quad (1)$$

donde $E = 2\pi\phi/\tau$. Se genera un operador unitario de tal modo que

$$U|\psi\rangle = e^{iH\tau}|\psi\rangle = e^{i2\pi\phi}|\psi\rangle. \quad (2)$$

A través de la acción controlada de U , se llega a un estado intermedio con registro R ,

$$|R\rangle \otimes |S\rangle = \left(\sum_n e^{i2\pi\phi n} \right) \otimes |\psi\rangle, \quad (3)$$

y posteriormente se aplica la transformación cuántica de Fourier inversa para obtener ϕ y se repiten todos los pasos, obteniendo cada vez mejores aproximaciones de la energía, hasta alcanzar la precisión deseada contra un valor de referencia.

3. Implementación del PEA

En el algoritmo PEA, el número de qubits requeridos escala linealmente con el número de funciones que describen la base del espacio de Hilbert y por lo tanto, el número de compuertas requeridas crece polinomialmente con el número de qubits. La función de onda del estado fundamental del sistema se representa en un registro de qubits S (estado) y otro registro de salida R que es usado para almacenar información intermedia del proceso de convergencia para obtener el valor propio E de la energía, en el caso en que el operador hermítico al que nos referimos sea el Hamiltoniano H .

El programa *PyQuante*[®] usado en [4] recibe la descripción de una molécula y genera el Hamiltoniano del sistema en forma matricial, calculando sus valores propios por el formalismo de Hartree-Fock o del funcional densidad. Este hamiltoniano es tomado por la rutina *Lime*, la cual exponencia el hamiltoniano, es decir, calcula $e^{iH\tau}$ por el método de los aproximantes de Padé. Posteriormente, un programa auxiliar denominado *Tabasco*, convierte la salida de *Lime* para construir la compuerta controlada $C-U$. Al final de este proceso tenemos una compuerta que calcula la Ec. (1) en cada paso del algoritmo de estimación de fase. Como ha sido indicado, es recomendable descomponer el espacio de Hilbert del cálculo en el producto de los subespacios de los qubits. Con este objetivo, la rutina *Sangrita* descompone la compuerta controlada en matrices o rotaciones base para generar un circuito o *algoritmo cuántico*. Con estas rutinas como insumo, el programa *Margarita* controla el proceso iterativo PEA hasta alcanzar la precisión requerida. La Figura 1 describe el proceso de estimación seguido por el código *TEQUILA*[®].



Figura 1. Representación esquemática del funcionamiento del programa Tequila®.

4. Ruido cuántico

Por *ruido cuántico* se entiende que el sistema de dos niveles que trabaja como qubit disipa energía hacia un ambiente. La interacción con ese ambiente es la responsable de los cambios o pérdidas en la información, haciendo que la unidad de información “el qubit” salga modificado después del proceso de interacción.

Se introduce ruido al cálculo tomando el qubit de salida al final de cada etapa del algoritmo PEA y sometiéndolo al efecto de ruido cuántico mediante operación de una compuerta unitaria U . Para modelar el sistema ambiente-qubit resulta conveniente asumirlo como un sistema cerrado y representar la operación ruidosa con una compuerta unitaria U . A esta operación entra un estado conjunto del sistema ambiente-qubit ($\rho \otimes \rho_{env}$) y sale un estado modificado ρ' . Se trata el ambiente como un reservorio, cuyas alteraciones en la interacción con el sistema qubit son despreciables. De este modo, el estado del ambiente sale al otro lado de la operación sin sufrir modificaciones.

Se puede suponer que el estado inicial del ambiente $\rho_{env} = |\rho_0\rangle\langle\rho_0|$ es cero en la base del sistema y que el estado del sistema modificado viene dado por el operador suma:

$$\rho' = \epsilon(\rho) = Tr_{env} [U(\rho_s \otimes \rho_{env})U^+],$$

$$\rho' = \sum_k \langle e_k | U \rho \otimes |e_0\rangle \langle e_0| U^+ |e_k\rangle. \quad (4)$$

Definimos un operador en el espacio de estados del sistema principal $E_k = \langle e_k | U |e_0\rangle$, tal que $\sum_k E_k E_k^+ \leq I$. Con esto, el estado de salida del sistema es igual a

$$\rho' = \sum_k E_k \rho E_k^+. \quad (5)$$

5. Corrección cuántica de errores

Para proteger la información contra los efectos del ruido cuántico, el qubit es codificado mediante el algoritmo de Shor, que consiste en la combinación adecuada de los algoritmos de corrección para fliepo de bit y de corrección para fliepo de fase [3], cuyos circuitos son ilustrados a continuación

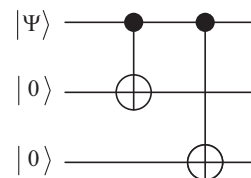


Figura 2. Circuito de corrección de fliepo de qubit. Las operaciones unitarias de dos qubits corresponden a la compuerta cuántica CNOT [3], donde el estado $|\Psi\rangle$ es el qubit de control.

El código de corrección de fliepo del qubit hace una especie de duplicación del qubit. Sea p la probabilidad

de que el qubit sea flipado, después de ser codificado como un bit triple *qubit lógico* [3],

$$\begin{aligned} |0\rangle &\rightarrow |0_l\rangle = |000\rangle, \\ |1\rangle &\rightarrow |1_l\rangle = |111\rangle, \end{aligned} \quad (6)$$

donde la probabilidad de error es $3p^2 - 2p^3$, la cual es mucho menor que p .

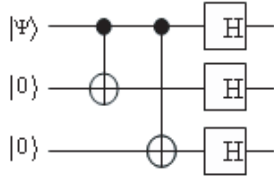


Figura 3. Circuito de corrección de flipeo de fase. H denota la compuerta de Hadamard; las operaciones de dos qubits corresponden a la compuerta CNOT [3], donde $|\Psi\rangle$ es el qubit de control.

El código de flipeo de fase trata este tipo de error como si fuese un flipeo de bit. Esto se puede hacer definiendo el cambio de base

$$\begin{aligned} |+\rangle &= \frac{(|0\rangle + |1\rangle)}{\sqrt{2}}, \\ |-\rangle &= \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}. \end{aligned} \quad (7)$$

En esta base, el operador de flipeo de fase σ_z actúa del mismo modo que el operador de flipeo de bit σ_x cambiando el qubit de $|+\rangle \rightarrow |-\rangle$ y viceversa. En esta nueva base, se propone la codificación

$$\begin{aligned} |0_l\rangle &= |+++ \rangle, \\ |1_l\rangle &= |-- - \rangle, \end{aligned} \quad (8)$$

la cual protege los qubits $|+\rangle$ y $|-\rangle$ contra el flipeo. Así, el circuito descrito anteriormente se encarga de hacer la duplicación de igual manera que el código de flipeo de qubit y posteriormente hace el cambio de variable mediante la operación de compuertas de Hadamard. La Figura 4 muestra el resultado de la combinación de estos dos circuitos, en lo que se conoce como el *algoritmo de Shor* [3].

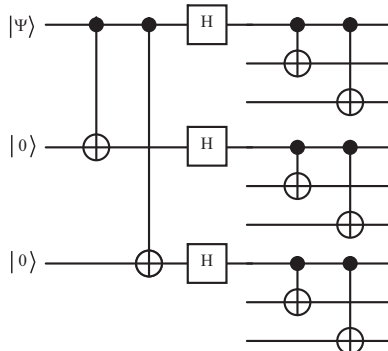


Figura 4. Esquema del algoritmo de Shor.

En nuestros cálculos numéricos este algoritmo ha sido considerado dentro de la estructura del programa *Tequila*® para la respectiva estimación de fase. Así extendemos el anterior programa al caso de procesos de ruido cuántico, como los definidos anteriormente.

Conclusiones. El método numérico del algoritmo PEA implementado en el programa *Tequila*® [4] posee precisión numérica hasta de 5 decimales. Las imprecisiones del método se adjudican a las aproximaciones hechas en la exponenciación del Hamiltoniano mediante aproximantes de Padé. La información cuántica puede ser protegida contra los errores de flipeo de fase, flipeo de bit y combinaciones de ambos, codificandolos mediante el algoritmo de Shor en un qubit robusto equivalente a 9 qubits, que al ser sometido a ruido cuántico no sufre los efectos de cambio o pérdida de la información contenida. Sin embargo, el costo de memoria y tiempo de cálculo sólo se incrementa polinomialmente en el cómputo cuántico (lo hace exponencialmente en la simulación clásica del mismo).

Agradecimientos. Este trabajo fue financiado parcialmente por Colciencias a través de los proyectos de investigación 1106-45-221296 y 1106-14-17903, y el programa de intercambio PROCOL (DAAD-Colciencias).

Referencias

- [1] P. W. Shor, Proceedings, 35th Annual Symposium on Foundations of Computer Science, IEE Press Los Alamitos CA (1994)
- [2] P. W. Shor, Proceedings, 37th Annual Symposium on Foundations of Computer Science, IEE Press Los Alamitos CA (1996)
- [3] M. A. Nielsen e I. L. Chuang, *Quantum Computation and Quantum Information*, Cambridge, Cambridge University Press (2000)
- [4] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, y M. Head-Gordon, *Science* **309**, 1704 (2005)
- [5] D. S. Abrams y S. Lloyd, *Phys Rev. Lett.* **79**, 2586 (1997)
- [6] D. S. Abrams y S. Lloyd, *Phys Rev. Lett.* **83**, 5162 (1999)