

Perhitungan Analitik Algoritma Grover

Suhadi*

*Dosen Pendidikan Fisika, Fakultas Ilmu Tarbiyah dan Keguruan, Universitas Islam Negeri Raden Fatah Palembang, 30126, Indonesia
suhadi@radenfatah.ac.id

Abstrak

Telah dilakukan perhitungan secara analitik ulang algoritma Grover pada komputasi kuantum untuk pencarian data dalam suatu basis data. Selain itu, dengan hasil juga diberikan hasil dari perhitungan numerik bahwa pencarian/query pada basis data data tak terurut dengan kompleksitas waktu $\mathcal{O}(\sqrt{N})$.

Latar Belakang

Perkembangan *hardware* komputer pada saat ini telah ditunjukkan oleh meningkatnya jumlah *microprocessor* yang ada pada sebuah komputer. Demikian yang ditunjukkan oleh Intel Corporation dalam https://www.intel.com/pressroom/kits/events/moores_law_40th/. Perkembangan *software* juga ditunjukkan dengan kemampuan komputasi sebuah komputer yang semakin cepat, meskipun komputer yang ada sekarang masih beroperasi berdasarkan paradigma klasik atau serial, sehingga sangatlah beralasan untuk mengembangkan sebuah komputer dengan paradigma kuantum, hal ini dikarenakan ruang Hilbert yang digunakan oleh mekanika kuantum sangatlah besar, sehingga beberapa permasalahan yang sulit pada komputer klasik dapat diselesaikan. Selain itu salah satu fitur yang ditawarkan oleh mekanika kuantum adalah prinsip superposisi yang memungkinkan evaluasi dari semua keadaan dilakukan pada waktu yang bersamaan. Fitur ini juga yang membuat komputer kuantum lebih *powerfull*.

Salah satu implementasi *Quantum Computing* sebagai sebuah model dalam paradigma komputasi yang baru adalah algoritma pencarian database dengan menggunakan algoritma Grover. Secara teori telah ditunjukkan bahwa pencarian dengan algoritma ini menunjukkan hasil yang signifikan dalam pencarian data pada sebuah database, untuk itulah pada penelitian ini bagaimanakah hasil dari simulasi dari algoritma tersebut terhadap pencarian sebuah data pada data-

base ?

Qubit

Data - data pada komputer direpresentasikan dalam bentuk bilangan biner yang disebut *bit*, yang bernilai 0 atau 1. Pada komputer klasik, kita dapat menuliskan sejumlah $2n$ kemungkinan dari data - data tersebut. Sebagai ilustrasi, misalkan data 5 bit dapat ditulis ke dalam 32 kemungkinan. Karena kinerja pada komputer klasik berjalan secara serial, maka pada setiap waktu kita hanya akan mendapatkan satu dari sejumlah kemungkinan yang ada. Sehingga, untuk semua kemungkinan, kita melakukan sejumlah $2n$ kali perhitungan. Berbeda dengan komputer klasik, pada komputer kuantum representasi data dituliskan dalam bentuk *state* vektor yang disebut *qubit* (*quantum bit*). Sebuah *qubit* dapat berada pada keadaan (*state*). sebuah qubit dapat berada pada keadaan (*state*) $|0\rangle$ atau $|1\rangle$, akan tetapi dapat juga berada pada keadaan superposisi

$$|\Psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (1)$$

dengan α dan β merupakan bilangan kompleks. Superposisi ini yang memungkinkan kita untuk menuliskan semua keadaan dalam satu kombinasi linier. Superposisi ini juga yang memungkinkan kita untuk mengevaluasi semua kemungkinan dalam satu kali perhitungan. Pada ruang vektor (1) dapat di-

tulis

$$\begin{aligned}
 |\Psi\rangle &= \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \\
 &= \begin{pmatrix} \alpha \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \beta \end{pmatrix} \\
 &= \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (2)
 \end{aligned}$$

$$\begin{aligned}
 H|1\rangle &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\
 &= \frac{1}{\sqrt{2}} \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \\
 &= \frac{1}{\sqrt{2}} \{|0\rangle - |1\rangle\} \quad (8)
 \end{aligned}$$

sehingga vektor basis $|0\rangle$ dan $|1\rangle$ masing-masing

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (3)$$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (4)$$

Sebuah *qubit* dapat direalisasikan dari polarisasi *photon*, mengarahkan spin inti yang berada pada medan magnet *uniform*, atau dari dua keadaan (*ground state* dan eksitasi) dari sebuah elektron yang mengorbit inti.

Gerbang Kuantum

Sebuah gerbang kuantum adalah analog dengan gerbang logika pada komputer klasik. Gerbang kuantum adalah komponen dasar dari algoritma kuantum. Perbedaan dalam konteks klasik dan kuantum adalah gerbang kuantum diimplementasikan secara uniter seperti

$$UU^\dagger = I \quad (5)$$

Salah satu gerbang kuantum yang sering digunakan adalah gerbang Hadamard

$$H \equiv \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (6)$$

yang biasa digunakan untuk membentuk superposisi dari dua keadaan seperti berikut

$$\begin{aligned}
 H|0\rangle &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\
 &= \frac{1}{\sqrt{2}} \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \\
 &= \frac{1}{\sqrt{2}} \{|0\rangle + |1\rangle\} \quad (7)
 \end{aligned}$$

Komputasi Kuantum dapat diimplementasikan berdasarkan serangkaian gerbang kuantum yang beroperasi secara uniter. Dalam mekanika kuantum operasi uniter ini diperlukan agar probabilitas sistem setelah operasi tidak berubah dan sifat ini digunakan sebagai dasar operasi berkebalikan pada Komputasi Kuantum selain prinsip superposisi yang menjadikan Komputasi Kuantum lebih powerful terhadap komputer klasik. Salah satu contoh sukses besar Komputasi Kuantum adalah algoritma Grover pada masalah pencarian data dari sejumlah entri tak terurut dari suatu basis data dengan kompleksitas waktu. Pada makalah ini kami tinjau ulang tentang algoritma Grover tersebut, dan dibuat aplikasi untuk mensimulasikan algoritma ini dalam bahasa pemrograman Java. Bagian teori makalah ini adalah tinjauan ulang mengenai algoritma Grover, bagian simulasi berisi tentang simulasi dalam bahasa pemrograman Java, dan bagian hasil dan diskusi menjelaskan hasil dari simulasi yang diperoleh.

Algoritma Grover

Pada pencarian sejumlah M data suatu basis data sepanjang $N = 2^n$ entri data tak terurut, algoritma Grover terdiri dari dua register. Register pertama merupakan superposisi semua kemungkinan state vektor, sementara register kedua merupakan register yang mengkodekan solusi. Secara sederhana, algoritma Grover dapat ditunjukkan sebagai berikut :

1. Siapkan sistem dalam keadaan superposisi
2. Lakukan Operasi Perulangan untuk operasi :

(a)

$$|\Psi'\rangle = O|\Psi\rangle \quad (9)$$

(b)

$$|\Psi''\rangle = D|\Psi'\rangle \quad (10)$$

3. Lakukan pengukuran terhadap $|\Psi''\rangle$

Perulangan akan berhenti jika $|\Psi''\rangle = |\xi\rangle$, jika tidak maka ulangi langkah 2.

Oracle O pada persamaan (9) didefinisikan sebagai

$$O \equiv I - 2|\xi\rangle\langle\xi| \quad (11)$$

yang berfungsi mem-flip-kan state vektor kepada solusi. Sementara operator D pada persamaan (10) didefinisikan seperti

$$D \equiv 2|\Psi\rangle\langle\Psi| - I \quad (12)$$

atau dapat ditulis dalam bentuk matrik

$$D = \begin{pmatrix} \frac{2}{N} - 1 & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N} - 1 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} - 1 \end{pmatrix} \quad (13)$$

Misalkan kita akan mencari sebuah data dengan index $|\xi\rangle = |2\rangle = (0 \ 1 \ 0)^T$ pada ruang vektor $N = 2^3 = 8$. Keadaan sistem dimulai dari

$$\begin{aligned} |0\rangle &= 1|000\rangle \\ &= 1 \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \end{aligned} \quad (14)$$

Penerapan gerbang Hadamard pada sistem memberikan hasil

$$\begin{aligned} |\Psi\rangle &= H|000\rangle \\ &= \frac{1}{2\sqrt{2}} \sum_{j=0}^7 |j\rangle \end{aligned} \quad (15)$$

untuk j merupakan nilai pada bilangan biner sehingga

$$\begin{aligned} |\Psi\rangle &= \frac{1}{2\sqrt{2}} \sum_{j=000}^{111} |j\rangle \\ &= \frac{1}{2\sqrt{2}} (|000\rangle + |001\rangle \\ &\quad + |010\rangle + |011\rangle + |100\rangle \\ &\quad + |101\rangle + |110\rangle + |111\rangle) \end{aligned} \quad (16)$$

Sesuai langkah pada Algoritma Grover maka Oracle yang kita gunakan adalah

$$\begin{aligned} O &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} - 2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \end{aligned} \quad (17)$$

Sehingga penerapan Oracle pada sistem memberikan

$$\begin{aligned} |\Psi'\rangle &= \frac{1}{2\sqrt{2}} (|000\rangle + |001\rangle \\ &\quad - |010\rangle + |011\rangle + |100\rangle \\ &\quad + |101\rangle + |110\rangle + |111\rangle) \end{aligned} \quad (18)$$

Setelah itu, kita kerjakan operator Difusi pada keadaan sistem terakhir

$$\begin{aligned} &[2|\Psi\rangle\langle\Psi| - I]|\Psi'\rangle = \\ &[2|\Psi\rangle\langle\Psi| - I] \left[|\Psi\rangle - \frac{2}{2\sqrt{2}} |010\rangle \right] \\ &= \frac{1}{2} |\Psi\rangle + \frac{1}{\sqrt{2}} |010\rangle \end{aligned} \quad (19)$$

Substitusi (16) pada (19) maka memberikan hasil

$$|\Psi''\rangle = \frac{1}{4\sqrt{2}} \sum_{\substack{j=0 \\ j \neq 2}}^7 |j\rangle + \frac{5}{4\sqrt{2}} |010\rangle. \quad (20)$$

Sampai di sini selesailah iterasi pertama dengan amplitudo solusi adalah $\left| \frac{5}{4\sqrt{2}} \right|^2 = 78.125$. Ulangi prosedur yang sama sehingga memberikan hasil

$$|\Psi''\rangle = -\frac{1}{8\sqrt{2}} \sum_{\substack{j=0 \\ j \neq 2}}^7 |j\rangle + \frac{11}{8\sqrt{2}} |010\rangle. \quad (21)$$

dengan amplitudo solusi adalah $\left| \frac{11}{8\sqrt{2}} \right|^2 = 94.53125$.

Kesimpulan

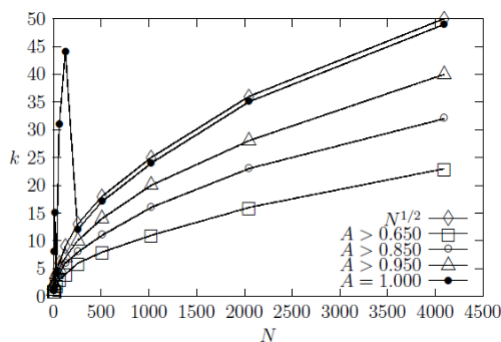
Algoritma kuantum merupakan algoritma yang dikembangkan berdasarkan paradigma mekanika kuantum. Algoritma Grover yang merupakan salah satu algoritma tersebut, memberikan hasil yang signifikan dalam suatu pencarian data. Selain perhitungan secara analitik sebagaimana diilustrasikan di atas, perhitungan numerik juga memberikan hasil yang signifikan sebagaimana pada uraian berikut :

Pada pencarian data dengan indeks $|3\rangle = (0 \ 0 \ 0 \ 1)^T$ pada basis data dengan jumlah data $N = 4$, dengan hasil bahwa state vektor yang dicari ditemukan pada iterasi ke 1. Hasil simulasi untuk jumlah data $N = 4$ sampai $N = 4096$ diberikan oleh Tabel.1. Tabel 1. Hasil simulasi algoritma Grover.

N	$\mathcal{O}(\sqrt{N})$	Iterasi Untuk :			
		$A^* > 0.650$	$A^* > 0.850$	$A^* > 0.950$	$A^* = 1.000$
4	2	1	1	1	1
8	2	1	1	2	8
16	3	1	2	2	15
32	4	2	3	4	4
64	6	3	4	5	31
128	9	4	6	7	44
256	13	6	8	10	12
512	18	8	11	14	17
1024	25	11	16	20	24
2048	36	16	23	28	35
4096	50	23	32	40	49

A*: Pembulatan Amplitudo sistem.

Adapun data pada tabel tersebut dapat dilihat pada Gambar 1.



Gambar 1 merupakan plot dari iterasi yang diperlukan pada simulasi algoritma Grover untuk berbagai pembulatan amplitudi state vektor yang dicari. Dari gambar terlihat untuk pembulatan $A > 0,65$, $A > 0,85$ dan $A > 0,95$ jumlah iterasi yang diperlukan lebih kecil dari $\mathcal{O}(\sqrt{N})$, akan tetapi untuk probabilitas sukses menemukan data 100%,

untuk jumlah data $4 \leq N < 128$ jumlah iterasi yang dibutuhkan lebih besar dari $\mathcal{O}(\sqrt{N})$, sedangkan untuk jumlah data $N \geq 256$ jumlah iterasi yang diperlukan $\approx \mathcal{O}(\sqrt{N})$. Dari hasil simulasi yang telah dilakukan dapatlah disimpulkan bahwa untuk kesuksesan pencarian 100% , algoritma Grover berhasil diterapkan untuk pencarian data dengan jumlah data yang besar khususnya untuk $N \geq 256$.

Pustaka

- [1] http://www.intel.com/pressroom/kits/events/moores_law_40th/
- [2] Yung Man-Hong, Whitfield J D., Boixo S, Tempel D G. and Aspuru-Guzik A, Introduction to Quantum Algorithms for Physics and Chemistry, arXiv:1203.1331v1 [quant-ph], 2012.
- [3] Das S and Kunstatter Q, Adiabatic quantum computation and Deutsch's algorithm, Phys. Rev. A 65, 062310, 2002.
- [4] Bang J, Yoo S, Lim J, Ryu J, Lee C and Lee J, Quantum heuristic algorithm for traveling salesman problem, arXiv:1004.4124v2 [quant-ph], 2010.
- [5] McMahon D, Quantum Computing Explained, John Wiley & Sons Inc. 2008.
- [6] Nielsen M A. and Chuang I L., Quantum Computation and Quantum Information 10th Anniversary Edition, Cambridge University Press, 2010.
- [7] Pinski S, Samson J, Adiabatic Quantum Computing, arXiv:1108.0560v1 [physics.pop-ph], 2011.
- [8] Dasgupta S., Papadimitriou C.H., and Vazirani U.V., Algorithms, 2006, <http://www.cs.berkeley.edu/~vazirani/algorithms.html>.
- [9] Barahona F, On the computational complexity of Ising spin glass models, J. Phys. A 15, 3241, 1982.
- [10] Shor P.W., Polynomial-Time Algorithms for Prime Factorization and Discrete

- Logarithms on a Quantum Computer, arXiv:quant-ph/9508027v2, 1996.
- [11] Harrow A W., Hassidim A and Lloyd S, Quantum Algorithm for Linear Systems of Equations, Phys. Rev. Lett, 103, 150502, 2009.
- [12] Pittenger A O., An Introduction to Quantum Computing Algorithms, Birkhäuser Boston, 2000.
- [13] Perry R T., The Temple of Quantum Computing Version 1.2, 2010.
- [14] Farhi E, Goldstone J, Gutmann S and Sipser M, Quantum Computation by Adiabatic Evolution, arXiv:quant-ph/0001106v1, 2000.
- [15] Cullimore M., Everitt M.J., Ormerod M. A., Samson J.H., Wilson R.D. and Zagoskin A.M., Relationship between minimum gap and success probability in adiabatic quantum computing, arXiv:1107.4034v2 [quant-ph], 2012.