Grover's algorithm and applications

by

Alain Tapp

CACR
Dept. of Combinatorics & Optimization
Faculty of Mathematics
University of Waterloo
Waterloo, Ontario, Canada N2L 3G1

email: atapp@cacr.math.uwaterloo.ca
http://www.iro.umontreal.ca/~tappa/

Plan

1. Grover's algorithm
   (a) NP
   (b) Grover's iteration
   (c) Search algorithm
   (d) applications
   (e) Heuristics
   (f) Optimality of the algorithm

2. Approximate counting
   (a) Grover's iteration + QFT = Counting
   (b) Counting algorithm and analysis
   (c) Severals accuracy levels
   (d) Applications

NP

A set S is in **NP** if there is a polynomial time algorithm F such that

$$\forall w \in S, \exists x, F_w(x) = 1$$

$$\forall w \notin S, \forall x, F_w(x) = 0$$

A set is **NPC** if it is in **NP** and every set in **NP** reduces to it in polynomial time.

Example of NPC problem

Scheduling:
Given a set of constraints C find a schedule s without conflicts. Thus $F_C(s) = 1$ iff s is a schedule without conflicts in C.

Travelling salesman:
Given a fixed budget c and the cost to travel between a list of cities C, give a tour t with cost less than the budget c. Thus $F_{(C,c)}(t) = 1$ iff t is an appropriate tour.

Knapsack:
Given a list of objects L with their weights and values, is it possible to get a subset with value at least v and with a total weight of w. Thus $F_{(L,v,w)}(s) = 1$ iff s is appropriate.

Satisfiability:
Given a Boolean expression E, give an assignment to the Boolean variables x_i such that $E(x_1, \ldots, x_n) = 1$. Thus $F_E(x) = 1$ iff $E(x) = 1$.

Search Problem

Searching a database

Given a table T and an entry y,
find i such that $T[i] = y$.

Searching under computable constraints

Given a boolean function $F : X \to \{0, 1\}$
find x such that $F(x) = 1$.

Note: It clearly relates to **NP** problems.

Grover's Iteration

$$\boxed{G_F = -HS_0HS_F}$$

$$S_0\,|i\rangle = \begin{cases} -\,|i\rangle & \text{if } i = 0 \\ |i\rangle & \text{otherwise.} \end{cases}$$

$$S_F\,|i\rangle = \begin{cases} -\,|i\rangle & \text{if } F(i) = 1 \\ |i\rangle & \text{otherwise.} \end{cases}$$

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

$$H^{\otimes n}\,|j\rangle = \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} (-1)^{i \cdot j}\,|i\rangle$$

Grover's Algorithm

Grover(F, m)

1. $|\Psi\rangle \leftarrow H\,|0\rangle$

2. Do m times
   $|\Psi\rangle \leftarrow G_F\,|\Psi\rangle$

3. Measure $|\Psi\rangle$ and output its value.

$$N = |X| \qquad t = |\{x \in X | F(x) = 1\}|$$

Success probability

Soufflé

Iteration analysis

$$|A\rangle = \sum_{F(x)=1} |x\rangle \qquad |B\rangle = \sum_{F(x)=0} |x\rangle$$

$$|A\rangle + |B\rangle \; = \; \sum_{x \in X} |x\rangle$$

$$\langle A|A\rangle \; = \; t \qquad \langle B|B\rangle \; = \; N - t$$

$$H\,|0\rangle \; = \; \frac{1}{\sqrt{N}} \sum_{x \in X} |x\rangle \; = \; \frac{1}{\sqrt{N}}\,|A\rangle + \frac{1}{\sqrt{N}}\,|B\rangle$$

Solve:

$$G_F^m(H\,|0\rangle) = k_m\,|A\rangle + \ell_m\,|B\rangle$$

Iteration analysis (2)

$$\begin{aligned}
G_F\,|\Psi\rangle &= -HS_0HS_F\;\;(k\,|A\rangle + \ell\,|B\rangle) \\
&= HS_0H\;\;(k\,|A\rangle - \ell\,|B\rangle) \\
&= H(I - 2\,|0\rangle\,\langle 0|)H\;\;(k\,|A\rangle - \ell\,|B\rangle) \\
&= (I - \frac{2}{N}(|A\rangle + |B\rangle)(\langle A| + \langle B|))\;\;(k\,|A\rangle - \ell\,|B\rangle) \\
&= k\,|A\rangle - \ell\,|B\rangle + \left(-\frac{2t}{N}k + 2\frac{N-t}{N}\ell\right)(|A\rangle + |B\rangle) \\
&= \left(\frac{N-2t}{N}k + \frac{2(N-t)}{N}\ell\right)|A\rangle \\
&\quad + \left(\frac{-2t}{N}k + \frac{N-2t}{N}\ell\right)|B\rangle
\end{aligned}$$

Iteration analysis

Theorem:
Let

$$\sin^2\theta = t/N$$

then

$$(G_F)^m(H\,|0\rangle) \;=\; k_m \sum_{F(x)=1} |x\rangle + \ell_m \sum_{F(x)=0} |x\rangle$$

where

$$k_m \;=\; \frac{\sin((2m+1)\theta)}{\sqrt{t}}$$

$$\ell_m \;=\; \frac{\cos((2m+1)\theta)}{\sqrt{N-t}}$$

When t is known

Theorem:
When

$$m \;=\; \lfloor\frac{\pi}{\text{arcsin}\left(\sqrt{t/N}\right)}\rfloor \in O(\sqrt{N/t})$$

Grover(F, m) outputs x such that $F(x) = 1$ with probability at least $\frac{N-t}{N}$.

Proof:
Just put the appropriate value of m in the amplitude equations of the previous slide.

When t is unknown

Theorem:
There exists a quantum algorithm **Search** that given F with $t > 0$ finds x such that $F(x) = 1$ with expected time in $O(\sqrt{N/t})$.

Search(F)

1. $m = 1, \lambda = 8/7$
2. $j \in_R \{0, \ldots, m-1\}$
3. $x =$**Grover**(F, j)
4. If $F(x) = 1$ then output x and stop
5. $m = min(\lambda, \sqrt{N})$
6. goto step 2.

Note: we can add a threshold of $O(\sqrt{N})$ if we are not sure that there is a solution.

Minimum

Theorem:
There exists an algorithm **Minimum** that finds x_0 such that $\forall x,\ F(x) \geq F(x_0)$, with probability 1/2, with an expected $O(\sqrt{N})$ calls to F.

Minimum(T)

1. $x_0 \in_R \{0, \ldots, N-1\}$
2. Define F such that $F(x) = 1 \Leftrightarrow T(x) < T(x_0)$
3. $x_1 =$**Search**(F)
4. If $T(x_1) < T(x_0)$ then $x_0 \leftarrow x_1$
5. If the cumulative number of calls to T is less than $25\sqrt{N}$ goto step 2
6. Output x_0.

Collision

Theorem:
Given $G : X \to Y$ a two-to-one function with $|X| = N$, the algorithm **Collision** finds (x_0, x_1) such that $G(x_0) = G(x_1)$ in time and space $O(\sqrt[3]{N})$.

Collision(T)

1. For i from 1 to $\sqrt[3]{N}$ set T[i]=(i,G(i)).

2. Sort T and look for collision in T

3. Define $F(x) = 1 \Leftrightarrow (x \geq \sqrt[3]{N}$ and $G(x) \in T)$

4. Set $x_0 =$**Search**(F) and x_1 such that $G(x_1) = G(x_0)$

5. Output (x_0, x_1).

Optimality

Theorem:
There is no algorithm that solves the problem **Search** with good probability with an expected number of call to F less than $\Omega(\sqrt{N})$.

Proof sketch:
Search start in state $|\Psi\rangle$ and call F via oracle O_x.

$$\begin{aligned}
|\Psi_k^x\rangle &= U_k O_x U_{k-1} \dots U_1 O_x |\Psi\rangle \\
|\Psi_k\rangle &= U_k U_{k-1} \dots U_1 |\Psi\rangle \\
D_k &= \sum_x \| \, |\Psi_k^x\rangle - |\Psi_k\rangle \, \|^2
\end{aligned}$$

Prove that:
1) D_K grows no faster than $O(k^2)$,
2) D_k must be in $\Omega(N)$ to distinguish N alternatives.

Examples of heuristics

Hill-Climbing: local variations that increase an objective function. Often very efficient!

Example: 3-Satisfiability, find assignment to $\{x_1, x_2, x_3, x_4\}$ that satisfies

$$(\bar{x_1} \vee \bar{x_4} \vee \bar{x_2})(\bar{x_1} \vee x_2 \vee \bar{x_3})(\bar{x_2} \vee \bar{x_4} \vee x_3)$$
$$(x_1 \vee \bar{x_1} \vee x_4)(x_4 \vee x_3 \vee x_3)(\bar{x_3} \vee \bar{x_4} \vee \bar{x_2})$$

Random assignment:
$x_1 = 1,\ x_2 = 1,\ x_3 = 1$ and $x_4 = 1$
satisfies 4 clauses
local variation $x_1 = 0$
satisfies 5 clauses
local variation $x_2 = 0$
satisfies all 6 clauses!

Heuristics

Let $\mathcal{F}$ be a family of functions of the form $F : X \to \{0, 1\}$ and $\mathcal{D}$ a probability distribution over this family.

A **heuristic** is a function $G : \mathcal{F} \times R \to X$.

Let $t_F = |\{x|F(x) = 1\}|$
and $h_F = |\{r|F(G(F, r)) = 1\}|$

A *good* heuristic is such that

$$E_{\mathcal{F}}\left(\frac{h_F}{|R|}\right) > E_{\mathcal{F}}\left(\frac{t_F}{|N|}\right)$$

Heuristics

Let $G'_F(r) = F(G(r, F))$

Algorithm:
Output $G(F,$ **Search**$(G'_F))$

Analysis:
Warning! In general

$$\left(\sum x_i\right)^{1/2} \leq \sum \sqrt{x_i}$$

but

$$\sum_{F \in \mathcal{F}} \sqrt{\frac{R}{t_F} P_F} = \sum_{F \in \mathcal{F}} \sqrt{\frac{R}{t_F} P_F} \sqrt{P_F} \leq$$

$$\left(\sum_{F \in \mathcal{F}} \frac{R}{t_F} P_F\right)^{1/2} \left(\sum_{F \in \mathcal{F}} P_F\right)^{1/2} = \left(\sum_{F \in \mathcal{F}} \frac{R}{t_F} P_F\right)^{1/2}$$

Approximate Counting

Counting Problem: given $F : X \to \{0,1\}$ with $|X| = N$ find $\tilde{t}$ a good estimate of $t = \{x | F(x) = 1\}$.

$\lvert t - \tilde{t} \rvert$	Quantum	Classical
$O(\sqrt{t})$	$O(\sqrt{N})$	$\Omega(N)$
ϵt	$O\left(\frac{1}{\epsilon}\sqrt{\frac{N}{t}}\right)$	$\Omega\left(\frac{N}{\epsilon^2 t}\right)$
< 1	$O(\sqrt{t(N-t)})$	$\Omega(N)$

Counting

The amplitude is a periodic function.
The period is related to t.

When m varies from 0 to $P-1$
k_m draws r periods of a sin function.

$$k_m \;=\; \frac{\sin((2m+1)\theta)}{\sqrt{t}}$$

$$r = P\theta/\pi$$

$$\sin^2(\theta) = \frac{t}{N}$$

Use Fourier analysis to evaluate r.

Basics Tools

Parameterize Grover's iteration

$$GI_F \;:\; |m\rangle \otimes |\Psi\rangle \rightarrow |m\rangle \otimes (G_F)^m\,|\Psi\rangle$$

Quantum Fourier Transform

$$QFT_P \;:\; |k\rangle \rightarrow \frac{1}{\sqrt{P}} \sum_{l=0}^{P-1} e^{2\pi i \frac{kl}{P}}\,|l\rangle \quad k \in Z_P$$

Note that:

$$QFT_P\,|0\rangle \;=\; \frac{1}{\sqrt{P}} \sum_{l=0}^{P-1} |l\rangle$$

Algorithm

Count(F, P)

1. $|\Psi_0\rangle \;\leftarrow\; |\mathbf{0}\rangle\, H^{\otimes n}\, |\mathbf{0}\rangle$

2. $|\Psi_1\rangle \;\leftarrow\; QFT_P \otimes I^{\otimes n}\, |\Psi_0\rangle$

3. $|\Psi_2\rangle \;\leftarrow\; GI_F\, |\Psi_1\rangle$

4. $|\Psi_3\rangle \;\leftarrow\; QFT_P^{-1} \otimes I^{\otimes n}\, |\Psi_2\rangle$

5. $\tilde{r} \;\leftarrow$ measure first register of $|\Psi_3\rangle$

6. Output: $\tilde{t} = N \sin^2 \frac{\tilde{r}\pi}{P}$ (and $\tilde{r}$ if needed)

Counting
Main Theorem

Theorem (Counting):
For $\tilde{t} = \textbf{Count}(F, P)$ then

$$|t - \tilde{t}| < \frac{2\pi}{P}\sqrt{t(N-t)} + \frac{\pi^2}{P^2}N$$

with probability at least $\frac{8}{\pi^2}$.

Proof

$$
\begin{aligned}
|\Psi_0\rangle &= \sum_{x \in X} \frac{1}{\sqrt{P}} |0\rangle |x\rangle \\
|\Psi_1\rangle &= \sum_{m=0}^{P-1} \sum_{x \in X} \frac{1}{\sqrt{PN}} |m\rangle |x\rangle \\
|\Psi_2\rangle &= \sum_{m=0}^{P-1} \frac{1}{\sqrt{P}} |m\rangle \left(k_m \sum_{F(x)=1} |x\rangle + \ell_m \sum_{F(x)=0} |x\rangle \right) \\
|\Psi_2\rangle &= \sum_{F(x)=1} \left(\sum_{m=0}^{P-1} \frac{k_m}{\sqrt{P}} |m\rangle \right) |x\rangle + \sum_{F(x)=1} \left(\sum_{m=0}^{P-1} \frac{\ell_m}{\sqrt{P}} |m\rangle \right) |x\rangle \\
\left|\Psi'_2\right\rangle &= \frac{1}{\alpha} \sum_{m=0}^{P-1} \sin((2m+1)\theta) |m\rangle \\
\left|\Psi'_3\right\rangle &= a\,|\lfloor r \rfloor\rangle + b\,|\lfloor r+1 \rfloor\rangle + c\,|P - \lfloor r \rfloor\rangle + \\
& \quad d\,|P - \lfloor r+1 \rfloor\rangle + |R\rangle
\end{aligned}
$$

Proof

With extensive algebraic manipulation, one can show that

$$| \, |R\rangle \, |^2 \; < \; 1 - \frac{8}{\pi^2},$$

thus with probability $\frac{8}{\pi^2}$ we have

$$\begin{aligned} |\tilde{r} - r| \; &< \; 1, \\ |\tilde{\theta} - \theta| \; &< \; \frac{\pi}{P}, \\ |\tilde{t} - t| \; &< \; \frac{2\pi}{P}\sqrt{t(N-t)} + \frac{\pi^2}{P^2}N. \end{aligned}$$

Good Estimation

Corollary 1:
Given F with N and t as defined before, **Count**$(F, c\sqrt{N})$ outputs $\tilde{t}$ such that

$$|t - \tilde{t}| < \frac{2\pi}{c}\sqrt{t} + \frac{\pi^2}{c^2}$$

with probability $\frac{8}{\pi^2}$ and requires exactly

$$c\sqrt{N}$$

evaluations of F.

Proof:
Replace P with $c\sqrt{N}$ in counting theorem.

Constant Factor Estimation

Corollary 2:
There exists an Algorithm **CountRel**(F, c) which output $\tilde{t}$ such that

$$|t - \tilde{t}| \;<\; \epsilon t$$

with probability 2/3 and runs in expected time

$$O\left(\frac{1}{\epsilon}\sqrt{N/t}\right).$$

CountRel(F, c)

1. $l = 0$
2. $l \leftarrow l + 1$
3. $\tilde{t} \leftarrow$**Count**$(F, 2^l)$
4. If $\tilde{t} = 0$ and $2^l < 2\sqrt{N}$ then goto step 2
5. Output **Count**$(F, \frac{200}{\epsilon}2^l)$

Probably Exact Counting

Corollary 3:
There exists an algorithm **Exact_Count** that output $\tilde{t}$ such that

$$\tilde{t} = t$$

with probability 2/3 and runs with expected time in

$$O(\sqrt{t(N-t)})$$

using only *constant space*.

Exact_Count(F)

1. $\tilde{t_1} \leftarrow$**Count**$(F, 50\sqrt{N})$ and $\tilde{t_2} \leftarrow$**Count**$(F, 50\sqrt{N})$
2. $P \leftarrow \mathrm{Min}(30\sqrt{\tilde{t_1}(N-\tilde{t_1})}, 30\sqrt{\tilde{t_1}(N-\tilde{t_1})})$
3. Output **Count**(F, P)

Other Applications (Sum)

Corollary 4:
Let $F : X \to Y$ with $|X| = N$ and Y an ordered set of n-bit numbers between 0 and 1 and let

$$S = \sum_{i \in X} F(i).$$

There exists an algorithm **Sum** that output $\tilde{S}$ such that

$$|\tilde{S} - S| < \sqrt{S}$$

which runs in time $O(n^2\sqrt{N})$.

Proof:
Let $F(i)_j$ be the jth bit of $F(i)$.

Sum(F, N, n) (Christoph Dürr 97)

1. $S \leftarrow 0$
2. For j ranging from 1 to n

   $S \leftarrow S + 2^j \textbf{Count}^n(F_j, \sqrt{N})$
3. Output S.

Other Applications (Selection)

Approximate Selection Problem:
Given $F : X \to Y$ with $|X| = N$ and k, find x_0 such that if $k' = |\{x | F(x) < F(x_0)\}|$ then $|k - k'| < 2\pi\sqrt{k} + \pi^2$.

Use binary search in combination with counting.

Can be solved in time $O(\log(N)^2\sqrt{N})$

References

Lov K. Grover, A fast quantum mechanical algorithm for database search, Proceedings of 28th Annual ACM Symposium on Theory of Computing, May 1996, pp. 212−219.
(quant-ph/9605043)

Michel Boyer, Gilles Brassard, Peter Høyer and Alain Tapp, Tight Bounds on Quantum Searching, Fortschritte der Physik, vol.46(4-5), 1998, pp. 493-505.
(quant-ph/9605034)

Gilles Brassard, Peter Høyer and Alain Tapp, Cryptology Column —Quantum Algorithm for the Collision Problem, ACM SIGACT News, Vol. 28, June 1997, pp. 14-19. Presented at LATIN'98.
(quant-ph/9705002)

References

Christoph Dürr and Peter Høyer, A Quantum Algorithm for Finding the Minimum
(quant-ph/9607014)

Gilles Brassard, Peter Høyer and Alain Tapp, Quantum Counting, 25th International Colloquium, ICALP'98, LNCS vol. 1443, Springer, pp.820-831,1998.
(quant-ph/9805082)

Gilles Brassard, Peter Høyer, Michele Mosca and Alain Tapp, Quantum Amplitude Amplification and Estimation, in Quantum Computation & Quantum Information Science, AMS Contemporary Math Series.
(quant-ph/0005055)