Lower Bounds for Quantum Algorithms on Independent Set and Subgraph Isomorphism

Matthew Mains

00089096

December 11, 2005

Abstract

This project is a basic explanation of the paper Quantum Algorithms and Lower Bounds for Independent Set and Subgraph Isomorphism Problem by S. Dörn in my own words, elaborating on and explaining points that were glossed over in the original paper. The algorithms have been kept intact, the proof of lower bounds have also not been fundamentally changed. The purpose of this paper is to give the lower bounds for computing graph problems, mainly independent set and subgraph isomorphism.

1 Introduction

The classical computer model, the Turing machine, is less than one hundred years old and already there is something better on the horizon. All classical computers are based on this model and hence are all subject to the same limitations. Quantum computers (and the Quantum Turing machine) seek to replace classical computers. One of the goals of studying quantum computers is to determine in which situations they can provide significant speed-up over the classical model. There are algorithms using quantum computers that show exponential improvement, consider Shor's factoring algorithm or Simon's problem. Even with such promise it is unlikely that a quantum Turing machine can solve all problems in NP efficiently (we define efficiently as solvable in polynomial time in the size of the input.)

This paper will summarize recently published results (see [1]) that show lower bounds for quantum algorithms machines in terms of query complexity and time complexity for several graph problems. We consider query complexity to be the number of queries made to an adjacency matrix of a graph, and the time complexity to be the number of steps a quantum algorithm would need in order to solve the problem. We will be considering maximal and maximum independent sets, and the subgraph isomorphism problem, though the focus of this paper is directed at the maximal and maximum independent set problems.

2 Complexity Classes

In computational complexity theory, one of the biggest unresolved problems is $P =^{?} NP$. We define a decision problem, R, to take an input x and the result is 'yes' if and only if there exists a y such that the property R(x,y) holds and 'no' otherwise. The set NP consists of all decision problems R the object y is polynomial size in terms of x, and the property R(x,y) can be verified in polynomial time on a classical Turing machine. The set P is the set of all decision problems that are solvable (and verifiable) in polynomial time. It is clear from the definitions that $P \subseteq NP$, however showing either $NP \nsubseteq P$ or $NP \subseteq P$ is an open problem. The hardest problems in NP are called NP-Complete (abbreviated NPC). Two of the three problems discussed in this paper are in NPC.

As mentioned previously a quantum algorithm for factoring integers was discovered by Shor [2]. Factoring has been assumed to be hard classically and thus is believed to be in NP. However, the factoring problem is not believed to be NP-Complete and hence it is believed that even quantum Turing machines will be unable to solve all problems in NP efficiently.

3 Graph Problems

3.1 Definition

A graph is an algebraic structure, G, defined by a set of vertices, V, and a relation on the vertices, E, where $E : V^2 \rightarrow \{0, 1\}$ is defined by $E(u, v)$ =1 if u,v are adjacent and 0 otherwise. We can visualize a graph as a set of nodes on a plane and edges as lines that connect the nodes, where there is a line joining two vertices u, v if and only if $E(u, v) = 1$. Another way to represent graphs is through an adjacency matrix. An adjacency matrix is a $|V| \times |V|$ boolean matrix M where $M_{uv} = 1$ if and only if $E(u, v) = 1$.

3.2 Independent Set

A subset V' of the vertices of a graph are defined to be independent if for all vertices $u, v \in V'$ with $u \neq v$, $E(u, v) = 0$. A maximal independent set is a subset V' of vertices that is independent and no other independent subset of vertices contains V' as a proper subset. A maximum independent set is the largest independent set of G. We can phrase finding the maximum independent set of a graph as a problem in NP as follows:

> **Input:** A graph $G = (V, E)$ and a number k
>
> **Output:** 'yes' if and only if is there an independent set of size k

We can easily see that this problem is in NP, and a fairly simple reduction from the NPC problem SAT reveals this problem to be in NPC. The related problem of maximal independent set is easier than maximum independent set since the maximum independent set is the largest set taken over all maximal independent sets. We will see a polynomial time quantum algorithm for finding maximal independent sets.

3.3 Subgraph Isomorphism

We begin by defining isomorphisms of graphs. An isomorphism is a bijection Φ from the vertex set of $G_1 = (V_1, E_1)$ to the vertex set $G_2 = (V_2, E_2)$ such that for all $u, v \in V_1$ such that $E_1(u, v) = 1$ if and only if $E_2(\Phi(u), \Phi(v)) = 1$. Therefore two graphs are

isomorphic if such an isomorphism exists. Visually we can imagine this as labeling the vertices in the first graph, and finding a labeling of the second graph that is consistent with the first. A subgraph of a graph $G = (V, E)$ is a graph $G' = (V', E')$ where $V' \subset V$ and $E'(u, v) = 1$ implies that $E(u, v) = 1$ for all $u, v \in V'$. We can state the subgraph isomorphism as a problem in NP in the following manner:

Input: Two graphs G_1, G_2

Output: 'yes' if and only if there is a subgraph of G_1 that is isomorphic to G_2

An interesting thing to note regarding the subgraph isomorphism problem is that it is an NPC problem; whereas finding if two graphs are isomorphic is believed not to be NPC (but is definitely in NP).

4 Algorithms and Analysis

4.1 Useful Theorems

We make use of Grover's search algorithm to find the ones in the adjacency matrix of a graph. In particular we use the following theorem:

Theorem: *Grover's search algorithm finds k items in a search space of size N in total time $O(\sqrt{kN})$. Additionally, to detect there are no more items in the search space takes an additional $O(\sqrt{N})$ time.*

So as a corollary we get the following result:

Corollary: *A search on an $N \times N$ boolean matrix, M, requires $O(N\sqrt{k})$ queries to M and outputs a set, S, of 1-entries ($M_{ij} = 1$) of M, where $|S| \geq k$ if M has more than k 1-entires, and all the 1-entires of M otherwise.*

This useful since we will be using the adjacency matrix model as an input to our quantum algorithms and having an upper bound on the time to find all the 1-entries of a matrix is useful in the analysis of the algorithms.

Since proving lower bounds for query complexity is difficult, we use a special case of a theorem by Ambainis [3] that reduces the amount of work to prove them significantly. The special case is as follows:

Theorem: *Let* $A, B \subset \{0,1\}^n$ *and* $f : \{0,1\}^n \to \{0,1\}$ *such that* $f(a) = 1$ *and* $f(b) = 0$ *for all* $a \in A$ *and* $b \in B$. *If*

1. *for all* $x = (x_1, x_2, \ldots, x_n) \in A$ *there exists at least* p *values* $i \in \{1, \ldots, n\}$ *such that* $(x_1, \ldots, 1 - x_i, \ldots, x_n) \in B$

2. *for all* $y = (y_1, y_2, \ldots, y_n) \in B$ *there exists at least* m *values of* $j \in \{1, \ldots, n\}$ *such that* $(y_1, \ldots, 1 - y_i, \ldots, y_n) \in A$

Then the minimum number of queries a bounded-error quantum algorithm must make to compute f *is* $\Omega(\sqrt{pm})$.

This theorem is useful since we can construct sets A,B as defined in the theorem for our graph problems and hence show the query lower bounds of the problems.

4.2 Maximal Independent Set

For solving the maximal independent set problem the paper proposed the following algorithm:

Input: A graph $G = (V, E)$ with an adjacency matrix A

Let $F = G$

Let $IS = \emptyset$

while vertex set of $F \neq \emptyset$

Take v any vertex from the vertex set of F

Set $IS = IS \cup \{v\}$

AV =search for all vertices adjacent to v in A using Grover

Remove AV from F and the edges adjacent to AV

It is clear this algorithm is correct since it removes all vertices adjacent to the chosen vertex at each step. Hence it will compute a maximal independent set for a given graph G, and chosen subset of vertices. However, it is also clear this will not necessarily compute the maximum independent set.

Claim: *The above algorithm makes* $O(n^{1.5})$ *queries to the adjacency matrix*

Proof: Using Grover's search to find adjacent vertex takes $O(\sqrt{n})$ queries to the adjacency matrix. We look at each vertex at most once, hence it follows that the algorithm makes $O(n * \sqrt{n}) = O(n^{1.5})$ queries. ■

Notice as well that the algorithm has a quantum time complexity of $O(n^{1.5})$. We can also prove a lower bound of $\Omega(n^{1.5})$ queries to the adjacency matrix for any quantum algorithm.

Claim: *Any quantum algorithm that solves the maximal independent set problem requires* $\Omega(n^{1.5})$ *queries to the adjacency matrix.*

Proof: We will construct sets A, B as required by the specialized version of Albainis's theorem.

Let A be the set of all graphs $G = (V, E)$ with $|V| = 3n+1$ with the following conditions:

1. There are n red vertices which are not adjacent to any other red vertex,
2. There are 2n green vertices not connected to any red vertices. Green vertices are grouped in pairs and each vertex in a pair is adjacent to the other vertex in the pair,
3. There is a black vertex which is adjacent to all red and green vertices.

Now we pick V' to be the set of all red vertices and n green vertices, one from each of the n green pairs. Then V' is a maximal independent set of G. We will say that $f(a) = 1$ for all $a \in A$.

Let B be the set of all graphs $G = (V, E)$ with $|V| = 3n + 1$ such that:

1. There are n+2 red vertices which are not adjacent to any other red vertex,
2. There are 2n-2 green vertices not adjacent any red vertices. Green vertices are grouped in pairs and each vertex in a pair is adjacent to the other vertex in the pair,
3. There is a single black vertex which is adjacent to all red and green vertices.

Notice that the V' from above is no longer a maximal independent set for a graph of B. So we set $f(b) = 0$ for all $b \in B$.

Now notice that forany graph $G' \in B$ we can obtain a graph $G \in A$ by adding an edge between red vertices (and re-colouring them green). So $l' = \frac{(n+2)(n+1)}{2} = O(n^2)$. Similarly, we can take any graph $G \in A$ and obtain $G' \in B$ by deleting an edge between 2 green vertices (and re-colouring them red). Hence $l = n = O(n)$.

Now by Ambainis's Theorem we have the query complexity to be $\Omega(\sqrt{l \cdot l'}) = \Omega(n^{1.5})$ as required. ■

Combining the results of the above two claims we can see that the presented algorithm is an optimal query algorithm for computing the maximal independent set of a graph, with query complexity $\Theta(n^{1.5})$.

4.3 Maximum Independent Set

On the surface it would appear that finding maximum and maximal independent sets of a graph are related problems. However, it is known that the maximum problem is in NPC, while there is a polynomial time algorithm that solves the maximal problem classically, hence it is in P and no reduction exists between the two. The key difference is a maximal set is a maximum independent set for a given set of vertices, whereas the maximum independent set of a graph is the largest of any such set. The proposed algorithm uses a classical probabilistic algorithm with quantum amplitude amplification. Consider the following algorithm:

Input: A graph $G = (V, E)$

Let $F = G$

Let $IS = \emptyset$

while the vertex set of $F \neq \emptyset$

  if max degree of F ≤ 2 then

    Set IS = maximum independent sets of all its components

    return IS

  Let v be vertex of maximum degree in F

  Let c be the result of flipping a fair coin

  if c is heads

    remove v from F

  otherwise

    add v to IS

    remove v and all neighbours of v from F

The above algorithm is a classical probabilistic algorithm to which we will apply quantum amplitude amplification to. In a paper by Brassard, Hóyer, Mosca and Tapp ([4]) the following was shown:

Theorem: *Let A be a quantum algorithm with one-sided error and success probability at least ϵ. Then, there is a quantum algorithm B that solves the same problem with success probability $\frac{2}{3}$ by invoking A $O(\frac{1}{\epsilon})$ times.*

We will now show the quantum time complexity of the algorithm:

Theorem: *The above algorithm combined with quantum amplitude amplification will compute a maximum independent set in expected time $O(2^{n/5}p(n))$ with constant probability. (where $p(n)$ is a polynomial function)*

Proof: Consider how the algorithm works, we begin by checking the maximum degree of vertices in the graph. If the maximum degree is two or less, we can find the maximum independent set in polynomial time (say $p(n)$) since the components of the graph are just cycles and paths. Otherwise we consider the case where the maximum degree of the graph is greater than two. We pick any vertex v with maximum degree and flip a fair coin. If the result is heads, we assume that v is not in the maximum independent set and delete it from F. If the result is tails, we assume v is in the maximum independent set, so we add it to IS and then delete v and all its neighbours (which obviously couldn't be in an independent set containing v) from F. Since $deg(v) \geq 3$ we delete at least 4 vertices each time we are in this case.

We must now consider how many iterations it will take for F to be empty. If x is the number of iterations required, we see that $n = \frac{1}{2}(x + 4x) \Rightarrow x = \frac{2n}{5}$. This is a rough estimate since at each iteration we remove either one or at least four vertices. Hence, we have

$$\epsilon = Prob(IS \text{ is a max. indep set}) \geq \left(\frac{1}{2}\right)^{\frac{2n}{5}}$$

Then applying the amplitude amplification we require $O(1/\sqrt{\epsilon}) = O(2^{\frac{2n}{5}})$ time to compute the maximum independent set of G. ■

Now that we have an upper bound for the runtime of a quantum algorithm we make use of Ambainis's theorem to help us prove a lower bound for the number of queries to an adjacency matrix is $\Omega(n^{1.5})$.

Claim: *The maximum independent set problem requires* $\Omega(n^{1.5})$ *queries to the adjacency matrix of a graph* $G = (V, E)$.

Proof: This proof will construct sets A,B as required by Ambainis's theorem. We use the same sets as before. Recall all graphs in A have n red vertices, 2n green vertices and 1 black vertex whereas graphs in B have n-2 red vertices, 2n+2 green vertices and 1 black vertex.

We can find the maximum independent set of $a \in A$ to be V', where V' is all the red vertices and one green vertex from each pair. Notice $|V'| = 2n$ for all graphs $a \in A$, so we let $f(a) = 1$.

If we construct a maximum independent set V'' of $b \in B$ we need to pick one green vertex from each pair and every red vertex. So $|V''| = n + 1 + n - 2 = 2n - 1$ for all $b \in B$, so we let $f(b) = 0$. Again applying Ambainis's theorem, we can deduce that $l = n = O(n)$ and $l' = O(n^2)$, and therefore we get the quantum query complexity to be $\Omega(n^{1.5})$. ■

We notice immediately that there is an exponential separation between the number of queries to the adjacency matrix and the runtime of the proposed quantum algorithm.

4.4 Graph Isomorphism

As previously mentioned there is an apparent separation between the graph isomorphism problem and the subgraph isomorphism problem. The subgraph isomorphism problem is known to be in NPC and the graph isomorphism problem is known to be in NP but no reduction exists that shows it to be in NPC.
We will begin our analysis of the graph isomorphism problem by giving a lower bound for the query complexity of any algorithm that solves the problem.

Claim: *The graph isomorphism problem requires* $\Omega(n^{1.5})$ *quantum queries to the adjacency matrix.*

Proof: As before, we construct A,B as per the previous proofs. Let $a \in A$, then a is isomorphic to all graphs $a' \in A$. So $f(a) = 1$ for all $a \in A$. Notice that a is not isomorphic to any graph $b \in B$, so $f(b) = 0$.
As before we conclude that the quantum query complexity is $\Omega(n^{1.5})$. ■

4.5 Subgraph Isomorphism

In this paper we study a special case of the subgraph isomorphism problem that involves finding a clique in a graph. A subset $V' \subseteq V$ of a graph $G = (V, E)$ is defined to be a clique if for every $u, v \in V'$ with $u \neq v$ we have $E(u, v) = 1$. Alternatively we can interpret this as V' forming the complete graph on $|V'|$ vertices. Additionally we will restrict our special case to finding the maximum clique in a graph.

Claim: *The maximum clique problem requires* $\Omega(n^{1.5})$ *queries to the adjacency matrix of a graph* $G = (V, E)$.

Proof: We give a proof by reducing the maximal independent set to finding a maximal clique.

We construct the graph $G' = (V, E')$, where $E'(u, v) = 1 - E(u, v)$. So our edge relation is now reversed; any two vertices adjacent in G are no longer adjacent and vice-versa. Notice now that any maximum independent set of G' will be a maximum clique for G. Since maximum independent set requires $\Omega(n^{1.5})$ quantum queries we have by reduction that maximum clique requires $\Omega(n^{1.5})$ quantum queries as required. ■

As a corollary to this, the subgraph isomorphism problem requires $\Omega(n^{1.5})$ quantum queries. We can perform a reduction from maximum clique to subgraph isomorphism, since maximum clique is a special case of subgraph isomorphism. Thus, it follows that such a reduction exists, and hence the corollary is true.

5 Conclusion

This paper show lower bounds, based on query complexity, for the graph problems of maximal and maximum independent set, as well as subgraph isomorphism. We found that the quantum query complexity for each of these problems is $\Omega(n^{1.5})$. For the problem of finding a maximal independent set, we gave an optimal query algorithm that solves the problem. However, in the case of finding a maximum independent set we gave an algorithm that took $O(2^{\frac{2n}{5}})$ quantum time.

References

[1] S. Dörn, *Quantum Algorithms and Lower Bounds for Independent Set and Subgraph Isomorphism Problem*, arXiv:quant-ph/0510084

[2] P. Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, SIAM Journal on Computing 26: pages 1484-1509, 1997

[3] A. Ambainis, *Quantum Lower Bounds by Quantum Arguments*, Journal of Computer and System Sciences 64: pages 750-767, 2002

[4] G. Brassard, P. Hóyer, M. Mosca, A. Tapp, *Quantum amplitude amplification and estimation*, Quantum Computation and QUantum Information: A Millennium Volume, AMS Contemporary Mathematics Series, 2000.