

Quantum Lower Bounds by Polynomials

Isaac & Mate

University of Cambridge

28/11/25

1 Introduction

Classical Querying

Review of Grover's Algorithm

Representing Boolean Functions as polynomial

Quantum Query Complexity

2 How to Prove Lower Bounds

3 Examples

1 Introduction

Classical Querying

Review of Grover's Algorithm

Representing Boolean Functions as polynomial

Quantum Query Complexity

2 How to Prove Lower Bounds

3 Examples

Classical Querying

- The goal is to compute $f(X)$ for a fixed function $f : \{0, 1\}^n \rightarrow \{0, 1\}$
- We have a black box that holds the input bitstring X and can extract a single bit of the input per query
- How many queries does it take to compute $f(X)$?
- Classically, bounding this problem is trivial.

1 Introduction

Classical Querying

Review of Grover's Algorithm

Representing Boolean Functions as polynomial

Quantum Query Complexity

2 How to Prove Lower Bounds

3 Examples

Review of Grover's Algorithm

- The goal is to find a specific marked item in an unstructured list.
- For n items, the complexity is $\mathcal{O}(\sqrt{n})$.
- The equivalent boolean function is the OR function.
- How can we be certain it cannot be done any better?

1 Introduction

Classical Querying

Review of Grover's Algorithm

Representing Boolean Functions as polynomial

Quantum Query Complexity

2 How to Prove Lower Bounds

3 Examples

Representing Boolean Functions as polynomial

- OR function

$$f(x, y, z) = x + y + z + xy + xz + yz + xyz$$

- Parity function

$$g(x, y, z, w) = 1 + x + y + z + w$$

- Majority function

$$h(a, b, c, d) = abc + abd + acd + bcd + abcd$$

Representing Boolean Functions as Exact Real Polynomials

- OR function

$$f(x_1 \dots x_n) = \sum_i x_i - \sum_{i < j} x_i x_j + \sum_{i < j < k} x_i x_j x_k - \dots + (-1)^{n-1} (x_1 \dots x_n)$$

- Parity function

$$g(x_1, \dots, x_n) = \frac{1 - (1 - 2x_1)(1 - 2x_2) \dots (1 - 2x_n)}{2}$$

1 Introduction

Classical Querying

Review of Grover's Algorithm

Representing Boolean Functions as polynomial

Quantum Query Complexity

2 How to Prove Lower Bounds

3 Examples

Quantum Query Complexity

- In the query model the input x is stored in random access memory with each access to the memory being unit cost.
- The quantum query complexity of some function f is the minimal number of queries needed for an algorithm which outputs the correct value of $f(x)$ (within some bound of error).
- In most cases, the overall computation time of a quantum query algorithm is close to the query complexity, making it a useful model.

- Formally, we model a query as the following quantum operation: $O_x : |i, b, w\rangle \rightarrow |i, b \oplus x_i, w\rangle$.
- α and β represent the amplitudes of the quantum state immediately before the query $|i, b, w\rangle = \alpha|i, 0, w\rangle + \beta|i, 1, w\rangle$
- The amplitude of $|i, 0, w\rangle$ after the query becomes $(1 - x_i)\alpha + x_i\beta$.
- The amplitude of $|i, 1, w\rangle$ after the query becomes $x_i\alpha + (1 - x_i)\beta$.

- Example where X : 1001

$$|\psi_{in}\rangle = |00\rangle_i |0\rangle_b |1\rangle_z + |01\rangle_i |0\rangle_b |1\rangle_z + |10\rangle_i |0\rangle_b |1\rangle_z + |11\rangle_i |0\rangle_b |1\rangle_z$$

- $x_{00} = 1, \quad x_{01} = 0, \quad x_{10} = 0, \quad x_{11} = 1$

$$|\psi_{out}\rangle = |00\rangle |\mathbf{1}\rangle |1\rangle + |01\rangle |\mathbf{0}\rangle |1\rangle + |10\rangle |\mathbf{0}\rangle |1\rangle + |11\rangle |\mathbf{1}\rangle |1\rangle$$

1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

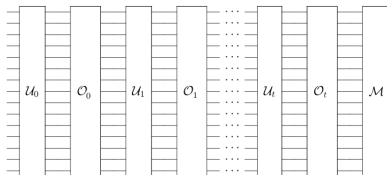
Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

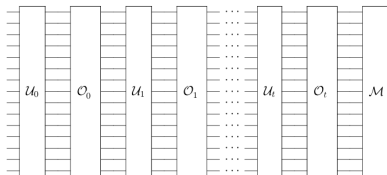
- Let \mathcal{A} be a quantum query algorithm making t queries to \mathcal{O}_w
- U_i denoting the i th unitary transform on the state.
- \mathcal{O}_w is the application of our oracle
- \mathcal{M} is the measurement



Theorem 1 (Final State Amplitudes)

- Let \mathcal{A} be a quantum query algorithm making t queries to black box.
- Then, there exist complex-valued multilinear polynomials $P_{i,b,z}(X)$, each of degree at most T , such that the final state is a superposition

$$|\psi_{final}\rangle = \sum_{i,b,z} P_{i,b,z}(X) |i, b, z\rangle$$



1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

- Base Case ($t = 0$)
- Before any queries are made, the algorithm prepares and applies unitary gates \mathcal{U}_0 that do not depend on X

$$|\psi_0\rangle = \sum_{i,b,z} \alpha_{i,b,z} |i, b, z\rangle$$

- The coefficients $\alpha_{i,b,z}$ are constants
- Constants are polynomials of degree 0

- Inductive Hypothesis ($t = k$)
- Assume that after k queries, the state is

$$|\psi_k\rangle = \sum_{i,b,z} P_{i,b,z}(X) |i, b, z\rangle$$

- The coefficients $\alpha_{i,b,z}$ are constant numbers
- where every amplitude $P_{i,b,z}(X)$ is a polynomial of degree at most k

- Inductive Step ($t = k + 1$)
- Now apply the $(k + 1)$ th query \mathcal{O}_x to the state $|\psi_k\rangle$

$$|\psi_k\rangle = \sum_{i,b,z} P_{i,b,z}(X) |i, b, z\rangle$$

- Using the formula derived earlier, the new amplitude $P'_{i,b,z}$ is

$$P'_{i,b,z}(X) = \underbrace{(1 - x_j) \cdot P_{i,b,z}(X)}_{\text{Contribution if } x_j=0} + \underbrace{x_j \cdot P_{i,b\oplus 1,z}(X)}_{\text{Contribution if } x_j=1}$$

- x_j has degree 1.
- Then $P(X)$ has degree $\leq k$ then $(x_j \cdot P) \leq k + 1$

- The Unitary Step

$$|\psi_{k+1}\rangle = U_{k+1}|\psi_{\text{post-query}}\rangle$$

- Since U_{k+1} is a linear transformation with constant coefficients, it mixes the amplitudes together but does not multiply them by x .
- Therefore adding polynomials of degree $k + 1$ return polynomial of degree $k + 1$

1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

From Amplitudes to Probabilities

Theorem 2 (Amplitudes-to-Probability Degree Relationship (Part 1/2))

- *Recap of Induction: We proved that after t queries, the amplitude of any basis state is a polynomial of degree at most t .*
- *Measurement Principle: In quantum mechanics, the probability of observing a state is the square of its amplitude:*

$$\text{Prob}(x) = |\alpha_x|^2$$

- *Polynomial Degree: Squaring a polynomial doubles its degree. Therefore, the acceptance probability $p(x)$ is a polynomial of degree at most $2t$.*

From Amplitudes to Probabilities

Theorem 3 (The Quantum Lower Bound (Part 2/2))

- *The Lower Bound Connection: If f requires a polynomial of degree at least d to approximate it (Approximate Degree), then our probability polynomial must satisfy:*

$$2T \geq d \implies T \geq \frac{d}{2}$$

Conclusion

The number of queries (T) must be at least half the approximate degree of the function.

1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

Representing Boolean Functions as Approximate Degree

- Exact degree requires perfect representation ($P(x) = f(x)$). But quantum computers are probabilistic.
- We only need $|P(x) - f(x)| \leq 1/3$.
- Let $f : \{0, 1\}^n \rightarrow \{0, 1\}$ be a Boolean function. The approximate degree $\deg(f)$ is the minimum degree of a real polynomial p such that:

$$|p(x) - f(x)| \leq \frac{1}{3} \quad \forall x \in \{0, 1\}^n$$

(Note: The polynomial operates over \mathbb{R} , allowing approximation errors.)

Representing Boolean Functions as Approximate Degree

For any Boolean function f , the quantum query complexity $Q(f)$ is lower bounded by the approximate degree:

$$\underbrace{\widetilde{\deg}(f)}_{\text{Approx Degree}} \leq 2 \cdot Q(f) \leq \underbrace{\deg(f)}_{\text{Exact Degree}}$$

1 Introduction

2 How to Prove Lower Bounds

Quantum Query Algorithm as a circuit

Proof by induction

Amplitudes to Probabilities

Approximate Degree

Symmetrization

3 Examples

Symmetrization of Polynomials

- For simplicity sake, we restrict our attention to symmetric functions, meaning $f(X) = f(\pi(X))$ for any permutation π .
- This means that f only depends on the hamming size of the input.
- Examples:
 - OR
 - AND
 - Parity
 - Majority

Symmetrization of Polynomials

Lemma 4 (Minsky, Papert)

Let $p : \mathbb{R}^n \rightarrow \mathbb{R}$ be a multilinear polynomial.

Then, there exists a polynomial $q : \mathbb{R} \rightarrow \mathbb{R}$ such that:

- ① $\deg(q) \leq \deg(p)$
- ② $p^{\text{sym}}(X) = q(|X|)$ for all $X \in \{0, 1\}^n$.

$$\underbrace{\widetilde{\deg}^{\text{sym}}(f)}_{\text{Symmetrized Approx}} \leq \underbrace{\widetilde{\deg}(f)}_{\text{Approx Degree}} \leq 2 \cdot \mathbf{Q}(f) \leq \underbrace{\deg(f)}_{\text{Exact Degree}}$$

1 Introduction

2 How to Prove Lower Bounds

3 Examples

Approximate Parity

Approximate OR

1 Introduction

2 How to Prove Lower Bounds

3 Examples

Approximate Parity

Approximate OR

Approximate Parity

- Let $f(x) = |x| \bmod 2$.
- Suppose the polynomial $p(x)$ approximates f , consider p^{sym} :

$$p^{\text{sym}}(0) \leq \varepsilon, p^{\text{sym}}(1) \geq 1 - \varepsilon, p^{\text{sym}}(2) \leq \varepsilon, \dots$$

- We see that $p^{\text{sym}}(x) - \frac{1}{2}$ has at least n roots, and thus $\deg(p) \geq n$. It follows that Parity requires at least $\frac{n}{2}$ queries.

1 Introduction

2 How to Prove Lower Bounds

3 Examples

Approximate Parity

Approximate OR

Approximate OR

Theorem 5 (Ehlich, Zeller; Rivlin, Cheney)

Let $p : \mathbb{R} \rightarrow \mathbb{R}$ be a polynomial such that:

$$\forall i \in \mathbb{N}. 0 \leq i \leq n \implies b_1 \leq p(i) \leq b_2$$

$$\exists x \in \mathbb{R}. 0 \leq x \leq n \wedge |p'(x)| \geq c$$

Then $\deg(p) \geq \sqrt{\frac{cn}{c+b_2-b_1}}$.

Approximate OR

- Let $f(x) = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x \neq 0 \end{cases}$
- Suppose the polynomial $p(x)$ approximates f , consider p^{sym} :

$$0 \leq p^{\text{sym}}(0) \leq \varepsilon$$

$$\forall i \in \mathbb{N}. 1 \leq i \leq n \implies 1 - \varepsilon \leq p^{\text{sym}}(i) \leq 1$$

$$\exists x \in \mathbb{R}. 0 \leq x \leq 1 \wedge (p^{\text{sym}})'(x) \geq 1 - 2\varepsilon$$

- Applying the theorem, we have $\deg(p^{\text{sym}}) \geq \sqrt{\frac{(1-2\varepsilon)n}{2-2\varepsilon}}$.