

# Quantum Lower Bounds Via Laurent Polynomials

**Justin Thaler (Georgetown University)**

with Scott Aaronson, Robin Kothari, William Kretschmer



# Query complexity

Let  $f: \{0,1\}^n \rightarrow \{0,1\}$  be a function and  $x \in \{0,1\}^n$  be an input to  $f$ .

$$x = \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & \cdots & x_n \\ \hline \end{array}$$

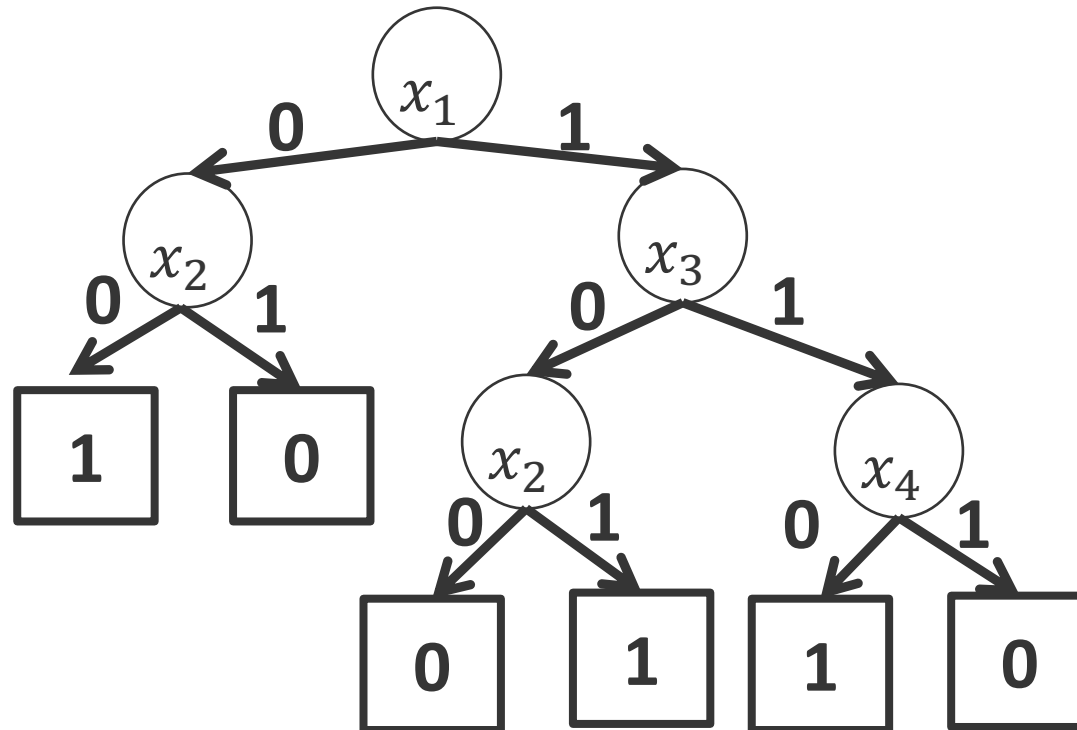
**Goal:** Compute  $f(x)$  by reading as few bits of  $x$  as possible.

# Query complexity

Let  $f: \{0,1\}^n \rightarrow \{0,1\}$  be a function and  $x \in \{0,1\}^n$  be an input to  $f$ .

$$x = \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & \cdots & x_n \\ \hline \end{array}$$

**Goal:** Compute  $f(x)$  by reading as few bits of  $x$  as possible.



# Query complexity

Let  $f: \{0,1\}^n \rightarrow \{0,1\}$  be a function and  $x \in \{0,1\}^n$  be an input to  $f$ .

$$x = \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & \cdots & x_n \\ \hline \end{array}$$

**Goal:** Compute  $f(x)$  by reading as few bits of  $x$  as possible.

**Quantum Query Complexity:** Algorithm can query bits of  $x$  in superposition, must output  $f(x)$  with probability at least  $2/3$ .

# Query complexity

Let  $f: \{0,1\}^n \rightarrow \{0,1\}$  be a function and  $x \in \{0,1\}^n$  be an input to  $f$ .

$$x = \begin{array}{|c|c|c|c|c|} \hline x_1 & x_2 & x_3 & \cdots & x_n \\ \hline \end{array}$$

**Goal:** Compute  $f(x)$  by reading as few bits of  $x$  as possible.

**Quantum Query Complexity:** Algorithm can query bits of  $x$  in superposition, must output  $f(x)$  with probability at least  $2/3$ .

**Example:** Let  $\text{OR}_n(x) = \bigvee_{i=1}^n x_i$  and  $\text{AND}_n(x) = \bigwedge_{i=1}^n x_i$ .

Then  $Q(\text{OR}_n) = Q(\text{AND}_n) = \Theta(\sqrt{n})$  [Grover96, Bennett-Bernstein-Brassard-Vazirani97]

Classically, we need  $\Theta(n)$  queries for both problems.

# Why query complexity?

## Complexity theoretic motivation

- We can prove statements about the power of different computational models! (E.g., exponential separation between classical and quantum algorithms)
- Oracle separations between classes, lower bounds on restricted models, upper and lower bounds in communication complexity, circuit complexity, data structures, etc.

## Algorithmic motivation

- Algorithms often transfer to the circuit model, while the abstraction of query complexity often gets rid of unnecessary details.
- Most quantum algorithms are naturally phrased as query algorithms. E.g., Shor, Grover, Hidden Subgroup, Linear systems (HHL), etc.

# Lower bounds on quantum query complexity

## Positive-weights adversary method [Ambainis]

Easy to use, but has many limitations. Cannot show any of the results of our work.

## Negative-weights adversary method [HLS07]

Equals (up to constants) quantum query complexity, but difficult to use.

In recent years, the adversary methods have become the tools of choice for proving lower bounds.

## Polynomial method

- Equals (up to constants) quantum query complexity for many natural functions.
- Can show lower bounds for algorithms with unbounded error, small error, and no error.
- Works when the positive-weights adversary fails (e.g., the collision problem).

# Lower bounds on quantum query complexity

## Positive-weights adversary method [Ambainis]

Easy to use, but has many limitations. Cannot show any of the results of our work.

## Negative-weights adversary method [HLS07]

Equals (up to constants) quantum query complexity, but difficult to use.

In recent years, the adversary methods have become the tools of choice for proving lower bounds.

## Polynomial method

- Equals (up to constants) quantum query complexity for many natural functions.
- Can show lower bounds for algorithms with unbounded error, small error, and no error.
- Works when the positive-weights adversary fails (e.g., the collision problem).
- Can imply lower bounds for **more powerful models** than quantum query complexity:
  - “Lifts” to quantum communication lower bounds [She08, SZ09]



# Lower bounds on quantum query complexity

## Positive-weights adversary method [Ambainis]

Easy to use, but has many limitations. Cannot show any of the results of our work.

## Negative-weights adversary method [HLS07]

Equals (up to constants) quantum query complexity, but difficult to use.

In recent years, the adversary methods have become the tools of choice for proving lower bounds.

## Polynomial method

- Equals (up to constants) quantum query complexity for many natural functions.
- Can show lower bounds for algorithms with unbounded error, small error, and no error.
- Works when the positive-weights adversary fails (e.g., the collision problem).
- Can imply lower bounds for **more powerful models** than quantum query complexity:
  - “Lifts” to quantum communication lower bounds [She08, SZ09]
  - **This work: Extensions to lower bound “super-powerful” query/communication models.**

# The Polynomial Method For Quantum Query Lower Bounds

**Approximate degree:** Minimum degree of a polynomial  $p(x_1, \dots, x_n)$  with real coefficients such that  $\forall x \in \{0,1\}^n, |f(x) - p(x)| \leq 1/3$ .

$\widetilde{\deg}(f)$

$$\widetilde{\deg}(\text{OR}_n) = \widetilde{\deg}(\text{AND}_n) = \Theta(\sqrt{n})$$

$$Q(\text{OR}_n) = Q(\text{AND}_n) = \Theta(\sqrt{n})$$

# The Polynomial Method For Quantum Query Lower Bounds

**Approximate degree:** Minimum degree of a polynomial  $p(x_1, \dots, x_n)$  with real coefficients such that  $\forall x \in \{0,1\}^n, |f(x) - p(x)| \leq 1/3$ .

$$\widetilde{\deg}(f)$$

$$\widetilde{\deg}(\text{OR}_n) = \widetilde{\deg}(\text{AND}_n) = \Theta(\sqrt{n})$$

$$Q(\text{OR}_n) = Q(\text{AND}_n) = \Theta(\sqrt{n})$$

Theorem ([Beals-Buhrman-Cleve-Mosca-de Wolf01]): For any  $f$ ,

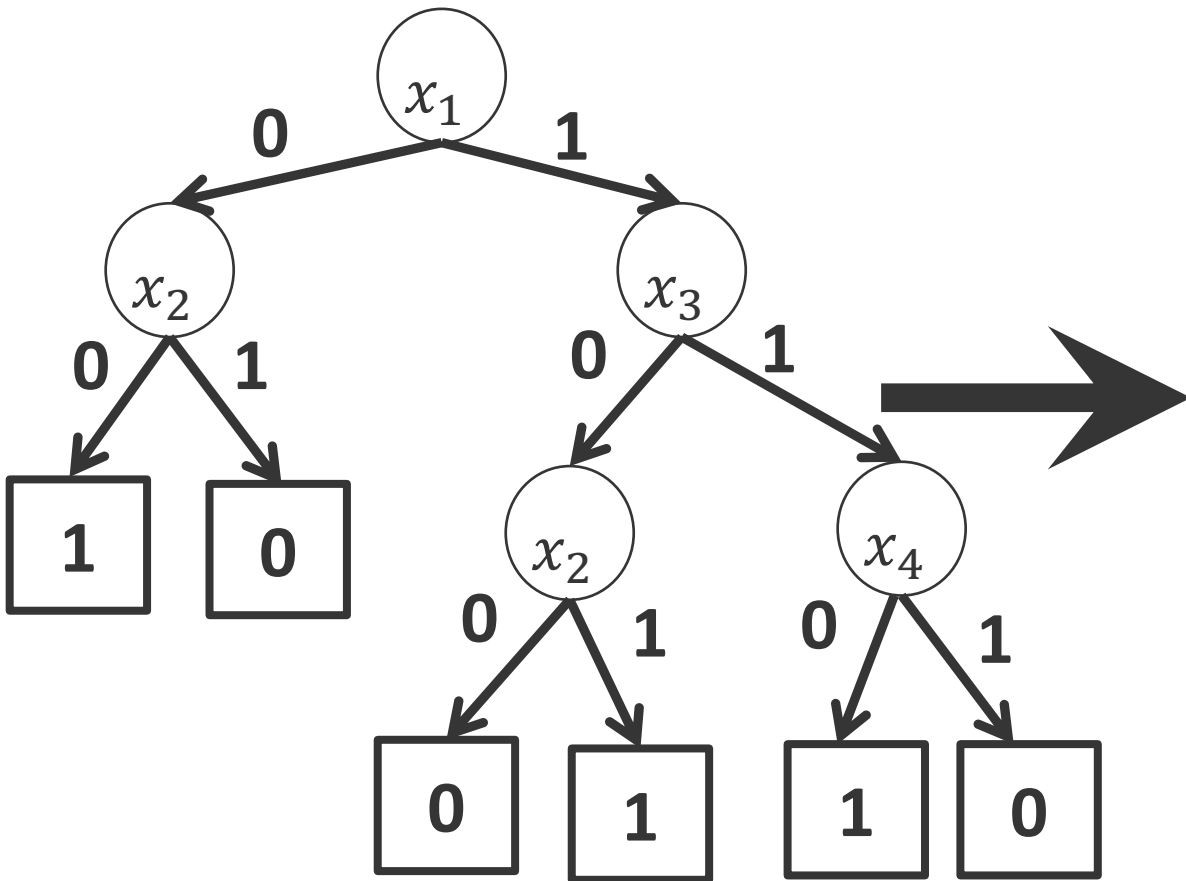
$$Q(f) \geq \frac{1}{2} \widetilde{\deg}(f)$$

The polynomial method

- For any  $T$ -query quantum algorithm  $A$ , there is a polynomial  $p$  of degree  $2T$  such that:
  - For all  $x \in \{0,1\}^n$ ,  $p(x)$  equals the probability that  $A$  outputs 1 on input  $x$ .

# Approximate degree and the Polynomial Method

- For any  $T$ -query quantum algorithm  $A$ , there is a polynomial  $p$  of degree  $2T$  such that:
  - For all  $x \in \{0,1\}^n$ ,  $p(x)$  equals the probability that  $A$  outputs 1 on input  $x$ .



$$p(x_1, x_2, x_3, x_4) = (1 - x_1)(1 - x_2) + x_1(1 - x_3)x_2 + x_1 x_3 x_4$$

# The Approximate Counting Problem

# Approximate Counting

- Given  $x \in \{0,1\}^n$ , let  $S = \{i : x_i = 1\}$ .

**Approximate counting problem ( $AC_{w,n}(x)$ ):** Determine whether  $|S| \leq w$  or  $|S| \geq 2w$ , promised that one of these is the case.

**Randomized query complexity:**

$$\theta(n/w)$$

**Quantum query complexity:**

$$\theta\left(\sqrt{n/w}\right)$$

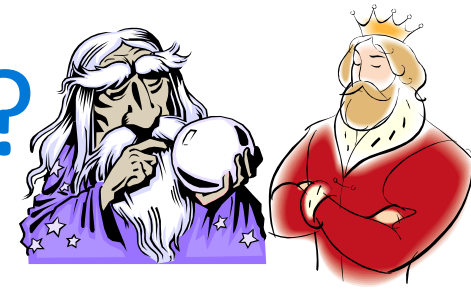
- Quantum Upper Bound (Brassard-Høyer-Tapp 1998):** Grover + quantum phase estimation (or just Grover...)
- Quantum Lower Bound (Nayak-Wu 1998):** Proven via polynomial method

# This Work: Understanding “Super-Powerful” Query Models

# First Result: QMA Protocols For Approximate Counting

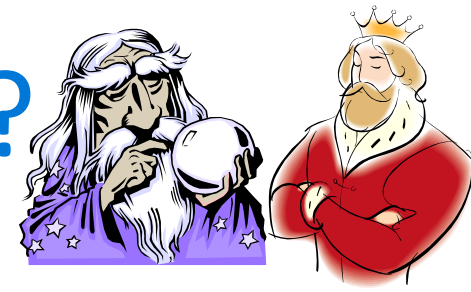


# QMA Protocol for Approximate Counting?



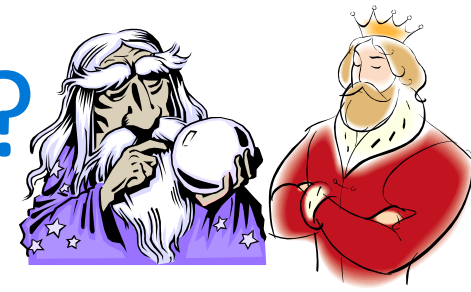
- In a QMA query protocol for  $f$ , Merlin knows the input  $x$  but Arthur does not.
- Merlin claims that  $f(x) = 1$ , and sends Arthur a **proof**  $|\varphi\rangle$  attesting to this.  $|\varphi\rangle$  is an arbitrary  $m$ -qubit message.
- After receiving  $|\varphi\rangle$ , Arthur queries at most  $T$  bits of the input in superposition.
- Completeness and soundness must hold.
  - $f(x) = 1 \implies$  there exists a  $|\varphi\rangle$  causing Arthur to accept with probability at least  $2/3$
  - $f(x) = 0 \implies$  for all possible proofs  $|\varphi\rangle$ , Arthur rejects with probability at least  $2/3$ .

# QMA Protocol for Approximate Counting?



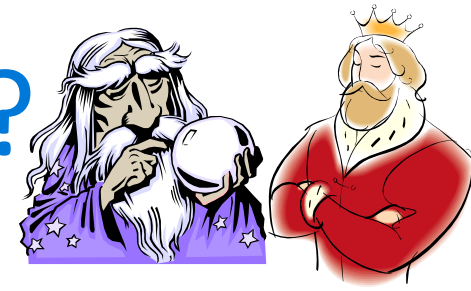
- In a QMA query protocol for  $f$ , Merlin knows the input  $x$  but Arthur does not.
- Merlin claims that  $f(x) = 1$ , and sends Arthur a **proof**  $|\varphi\rangle$  attesting to this.  $|\varphi\rangle$  is an arbitrary  $m$ -qubit message.
- After receiving  $|\varphi\rangle$ , Arthur queries at most  $T$  bits of the input in superposition.
- Completeness and soundness must hold.
  - $f(x) = 1 \implies$  there exists a  $|\varphi\rangle$  causing Arthur to accept with probability at least  $2/3$
  - $f(x) = 0 \implies$  for all possible proofs  $|\varphi\rangle$ , Arthur rejects with probability at least  $2/3$ .
- **Cost** of a protocol is the length  $m + T$ .

# QMA Protocol for Approximate Counting?



- In a QMA query protocol for  $f$ , Merlin knows the input  $x$  but Arthur does not.
- Merlin claims that  $f(x) = 1$ , and sends Arthur a **proof**  $|\varphi\rangle$  attesting to this.  $|\varphi\rangle$  is an arbitrary  $m$ -qubit message.
- After receiving  $|\varphi\rangle$ , Arthur queries at most  $T$  bits of the input in superposition.
- Is there an efficient QMA protocol for Approximate Counting?
  - i.e., Arthur is promised that either  $|S| \leq w$  or  $|S| \geq 2w$ , and Merlin wants to **prove** that  $|S| \geq 2w$ .
  - “Efficient” means cost  $\text{polylog}(n)$ .

# QMA Protocol for Approximate Counting?



- In a QMA query protocol for  $f$ , Merlin knows the input  $x$  but Arthur does not.
- Merlin claims that  $f(x) = 1$ , and sends Arthur a **proof**  $|\varphi\rangle$  attesting to this.  $|\varphi\rangle$  is an arbitrary  $m$ -qubit message.
- After receiving  $|\varphi\rangle$ , Arthur queries at most  $T$  bits of the input in superposition.
- Is there an efficient QMA protocol for Approximate Counting?
  - i.e., Arthur is promised that either  $|S| \leq w$  or  $|S| \geq 2w$ , and Merlin wants to **prove** that  $|S| \geq 2w$ .
- Obvious solutions:
  1. Merlin sends  $2w$  elements of  $S$ . Arthur picks a constant number of them and confirms they are all in  $S$  with one membership query each. Cost is  $O(w)$ .
  2. Arthur ignores Merlin and solves the problem with  $O(\sqrt{n/w})$  queries.

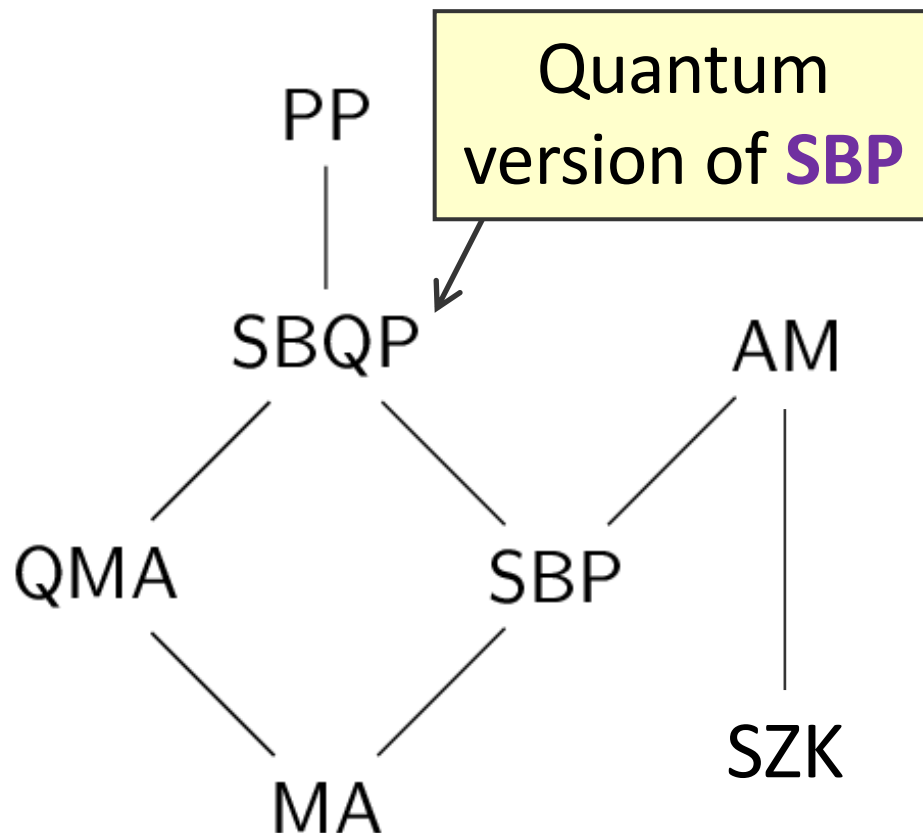
# Our Result

**Theorem:** Given  $S \subseteq [n]$ , for any QMA protocol for Approximate Counting that uses  $T$  queries to  $S$  and an  $m$ -qubit witness, either:

$$m \geq \Omega(w) \text{ or } T \geq \Omega(\sqrt{n/w}).$$

# Corollary: An Oracle Separating SBP and QMA

**SBP**: Class of languages  $L$  for which there's a polytime randomized algorithm that, for some  $\varepsilon$ , accepts w.p.  $\geq 2\varepsilon$  if  $x \in L$ , or w.p.  $\leq \varepsilon$  if  $x \notin L$ .



**Problem that had been open:** Is there an oracle relative to which

**SBP**  $\not\subseteq$  **QMA** ?

**Known oracle separations:**

**coNP**  $\not\subseteq$  **QMA** (easy)

**AM**  $\not\subseteq$  **PP** (Vereshchagin'92)

**SZK**  $\not\subseteq$  **QMA** (A. 2010)

# Background on QMA lower bounds

- [Vyalıi 2003, Marriott and Watrous 2005]: Any QMA query protocol for a function  $f$  with proof length  $m$  and query cost  $T$  can be transformed into a (Merlin-less) quantum query protocol  $Q$  of cost  $O(mT)$  satisfying:
  - $f(x) = 1 \implies \Pr[Q \text{ accepts } x] \geq 2^{-m}$
  - $f(x) = 0 \implies \Pr[Q \text{ accepts } x] \leq 2^{-m-1}$
- In complexity-theoretic terms,  $\text{QMA} \subseteq \text{SBQP}$ .

# Background on QMA lower bounds

- [Vyalyi 2003, Marriott and Watrous 2005]: Any QMA query protocol for a function  $f$  with proof length  $m$  and query cost  $T$  can be transformed into a (Merlin-less) quantum query protocol  $Q$  of cost  $O(mT)$  satisfying:
  - $f(x) = 1 \implies \Pr[Q \text{ accepts } x] \geq 2^{-m}$
  - $f(x) = 0 \implies \Pr[Q \text{ accepts } x] \leq 2^{-m-1}$
- In complexity-theoretic terms,  $\text{QMA} \subseteq \text{SBQP}$ .
- Major challenge to QMA lower bounds for  $\text{AC}_{w,n}$ :
  - $\text{AC}_{w,n}$  has a trivial SBP protocol  $Q$  of low cost.
  - $Q$  picks a random  $i \in [n]$ , queries  $x_i$ , and accepts if  $x_i=1$ .
  - $\text{AC}_{w,n}(x) = 1 \implies \Pr[Q \text{ accepts } x] \geq \frac{2w}{n}$
  - $\text{AC}_{w,n}(x) = 0 \implies \Pr[Q \text{ accepts } x] \leq \frac{w}{n}$



# Getting To Know Approximate Counting and the Polynomial Method

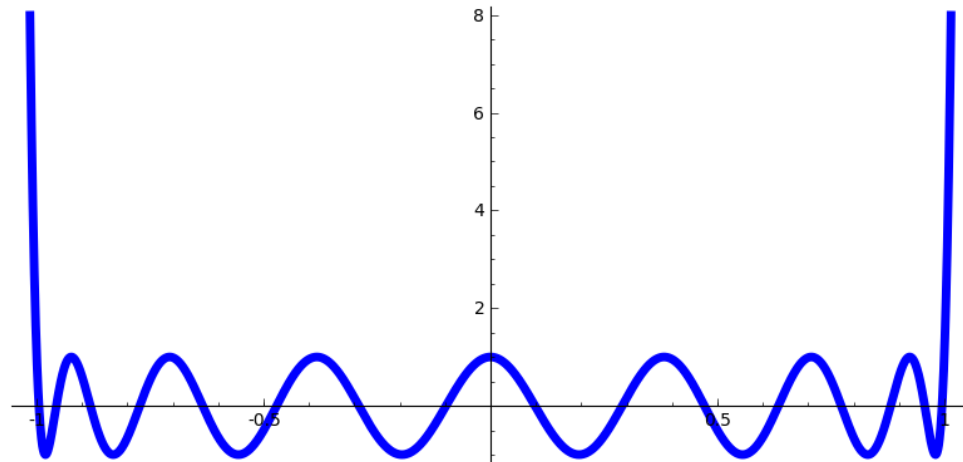
# The Approximate Degree of $\text{AC}_{w,n}$ (Upper Bound)

$$\widetilde{\deg}(\text{AC}_{w,n}) = \Theta(\sqrt{n/w}).$$

- Upper bound: Use **Chebyshev Polynomials**.
- Markov's Inequality: Let  $G(t)$  be a univariate polynomial s.t.  $\deg(G) \leq d$  and  $\max_{t \in [-1,1]} |G(t)| \leq 1$ . Then

$$\max_{t \in [-1,1]} |G'(t)| \leq d^2.$$

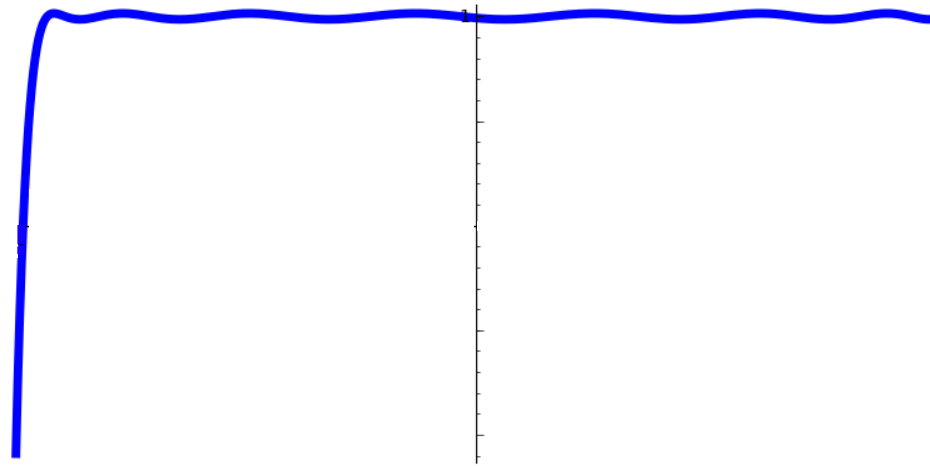
- Chebyshev polynomials are the extremal case.



# The Approximate Degree of $AC_{w,n}$ (Upper Bound)

$$\widetilde{\deg}(AC_{w,n}) = O(\sqrt{n/w}).$$

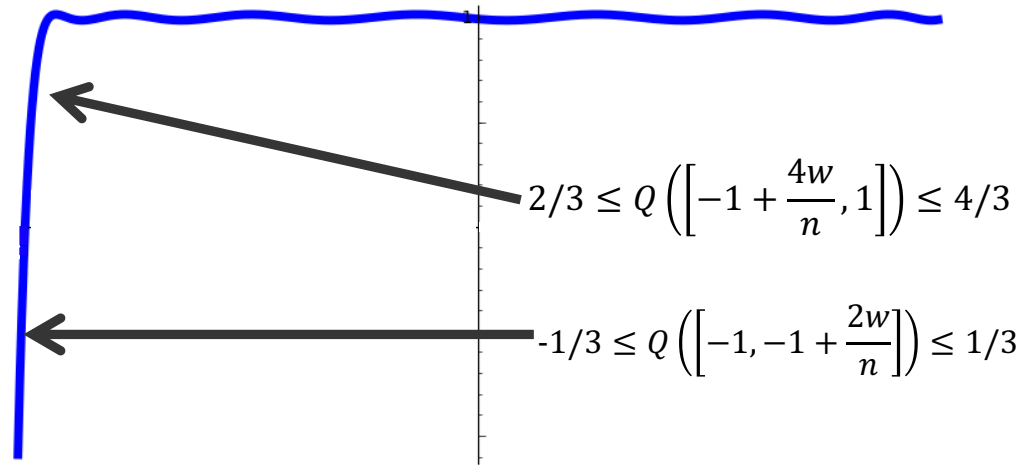
- After shifting and scaling, can turn degree  $O(\sqrt{n/w})$  Chebyshev polynomial into a univariate polynomial  $Q(t)$  that looks like:



# The Approximate Degree of $AC_{w,n}$ (Upper Bound)

$$\widetilde{\deg}(AC_{w,n}) = O(\sqrt{n/w}).$$

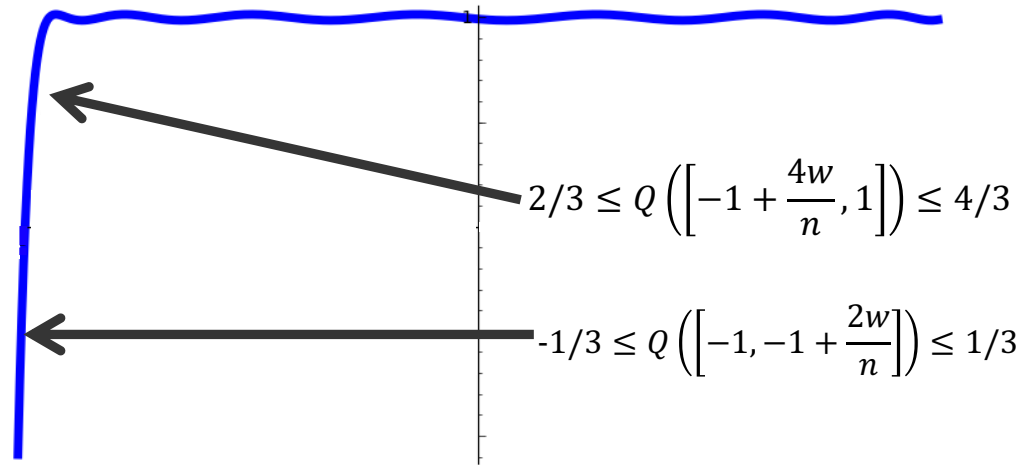
- After shifting and scaling, can turn degree  $O(\sqrt{n/w})$  Chebyshev polynomial into a univariate polynomial  $Q(t)$  that looks like:



# The Approximate Degree of $\text{AC}_{w,n}$ (Upper Bound)

$$\widetilde{\deg}(\text{AC}_{w,n}) = O(\sqrt{n/w}).$$

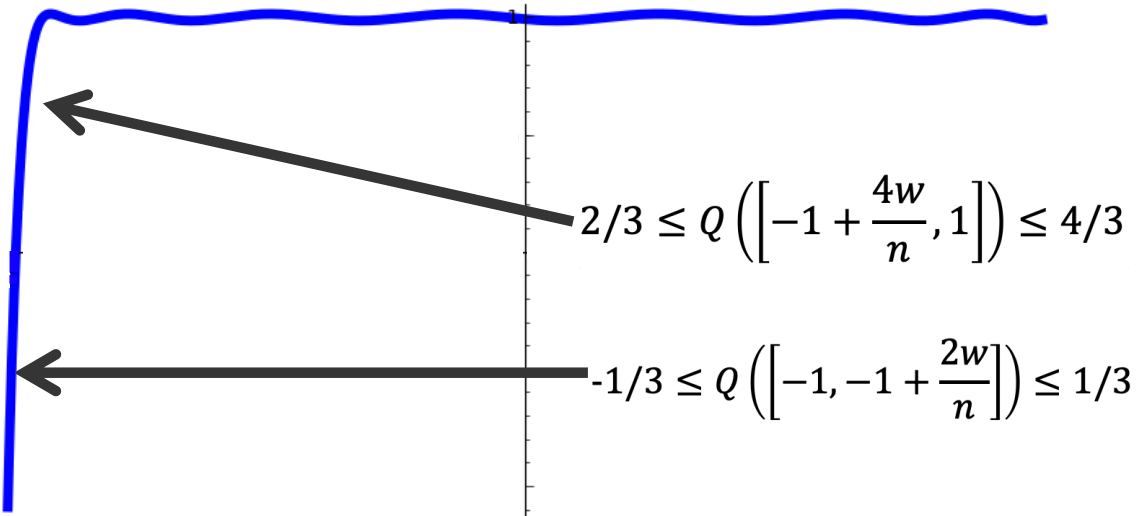
- After shifting and scaling, can turn degree  $O(\sqrt{n/w})$  Chebyshev polynomial into a univariate polynomial  $Q(t)$  that looks like:



- Define  $n$ -variate polynomial  $p$  via  
$$p(x) = Q(\sum_{i=1}^n (1 - 2x_i)/n).$$
- Then  $|p(x) - \text{AC}_{w,n}(x)| \leq 1/3 \quad \forall x \in \{0, 1\}^n.$

# The Approximate Degree of $AC_{w,n}$ (Lower Bound)

[NS92, NW98]  $\widetilde{\deg}(AC_{c,n}) = \Omega(\sqrt{n/w})$ .

- Lower bound: Use **symmetrization**.
  - Suppose  $|p(x) - AC_{w,n}(x)| \leq 1/3 \quad \forall x \in \{0, 1\}^n$ .
  - There is a way to turn  $p$  into a univariate polynomial  $p^{\text{sym}}$  that looks like this:
- 

- Claim 1:  $\deg(p^{\text{sym}}) \leq \deg(p)$ .
- Claim 2: Markov's inequality  $\implies \deg(p^{\text{sym}}) = \Omega(\sqrt{n/w})$ .

# What is $p^{\text{sym}}$ ?

**Theorem (Minsky and Papert, 1969):** Given a polynomial  $p(x_1, \dots, x_n)$  of total degree  $d$ , there exists a degree  $d$  univariate polynomial  $p^{\text{sym}}$  such that for all integers  $i = 0, \dots, n$ ,

$$p^{\text{sym}}\left(\frac{i}{n}\right) = \mathbf{E}_{|x|=i}[p(x)].$$

# What is $p^{\text{sym}}$ ?

**Theorem (Minsky and Papert, 1969):** Given a polynomial  $p(x_1, \dots, x_n)$  of total degree  $d$ , there exists a degree  $d$  univariate polynomial  $p^{\text{sym}}$  such that for all integers  $i = 0, \dots, n$ ,

$$p^{\text{sym}}\left(\frac{i}{n}\right) = \mathbf{E}_{|x|=i}[p(x)].$$

- Note: For inputs  $j \in [0,1]$  that are **not** integer multiples of  $1/n$ ,  $|p^{\text{sym}}(j)|$  can be as large as  $2^{d^2/n}$  [Coppersmith Rivlin 1992, BuhrmanClevedeWolfZalka 1999].
  - Not a worry if the degree lower bound to be shown is no larger than  $\sqrt{n}$ , since then  $2^{d^2/n} = O(1)$ .



# Summary: Quantum Query Lower Bound for $AC_{w,n}$

1. Start with any  $T$ -query quantum algorithm for  $AC_{w,n}$ .
2. Turn it into a degree- $(2T)$  polynomial  $p(x_1, \dots, x_n)$  approximating  $AC_{w,n}$ .
3. Turn  $p$  into a degree- $(2T)$  **univariate** polynomial  $p^{\text{sym}}$  that on input  $\frac{i}{n}$  outputs  $p$ 's average value on input sets  $S$  of size  $i$ .
4. Conclude that  $\deg(p^{\text{sym}}) \geq \Omega(\sqrt{n/w})$  and hence  $T \geq \Omega(\sqrt{n/w})$ .

# Proof of Result 1: QMA Lower bound for $AC_{w,n}$

# Laurent Polynomials

- Both of our results require generalizing the usual polynomial method to **Laurent polynomials**—although for different reasons in the two cases.

$$p(x) = 3x^{10} - x^4 + 1.5x + 7 - 2.2x^{-1} + x^{-5}$$

**Degree 10**

**Antidegree 5**

# QMA Lower Bound Attack Plan

**Recall Key Difficulty:** All known techniques for putting black-box problems outside **QMA**, also put them outside the larger class **SBQP**. But clearly no **SBP** problem can be outside **SBQP**!

**Key Idea of Thomas Watson:** **QMA** is closed under intersection! So suppose  $\text{SBP} \subseteq \text{QMA}$ . Then for all  $L_1, L_2 \in \text{SBP}$ , we'd also have  $L_1 \cap L_2 \in \text{QMA} \subseteq \text{SBQP}$ .

Therefore, we just need to show that the AND of two black-box  $\text{AC}_{w,n}$  instances is **not** in **SBQP**. This will contradict the assumption  $\text{SBP} \subseteq \text{QMA}$ .

- Thus, consider a **SBQP** algorithm for **two** approximate counting instances, on  $S \subseteq [n]$  and  $T \subseteq [n]$ :

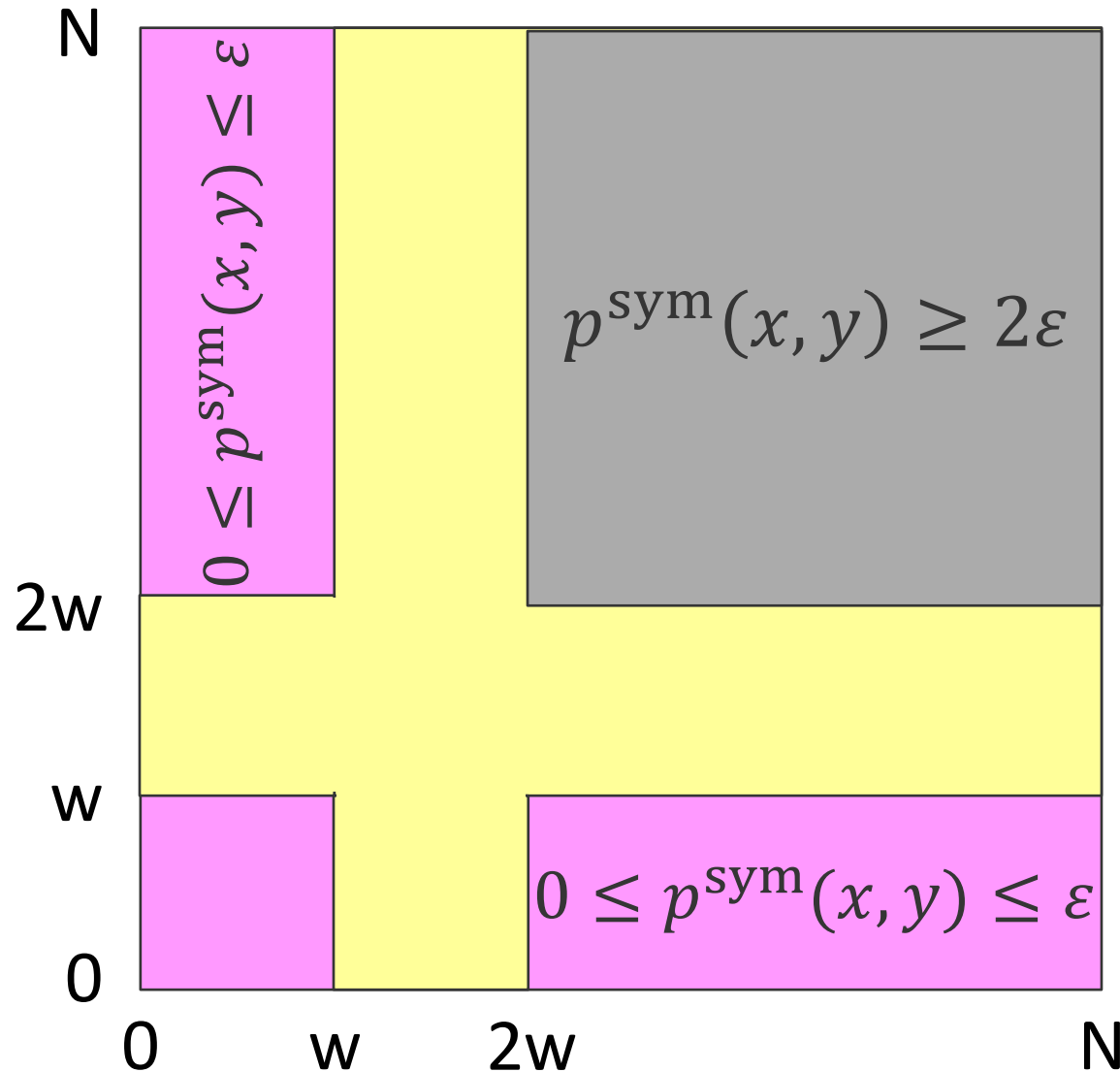
$$AC_{w,n}(S) \wedge AC_{w,n}(T)$$

- Let  $p(S, T)$  be its acceptance probability. After “double symmetrization,” we get a bivariate real polynomial

$$p^{\text{sym}}(x, y) = \mathbf{E}_{|S|=x, |T|=y} [p(S, T)].$$

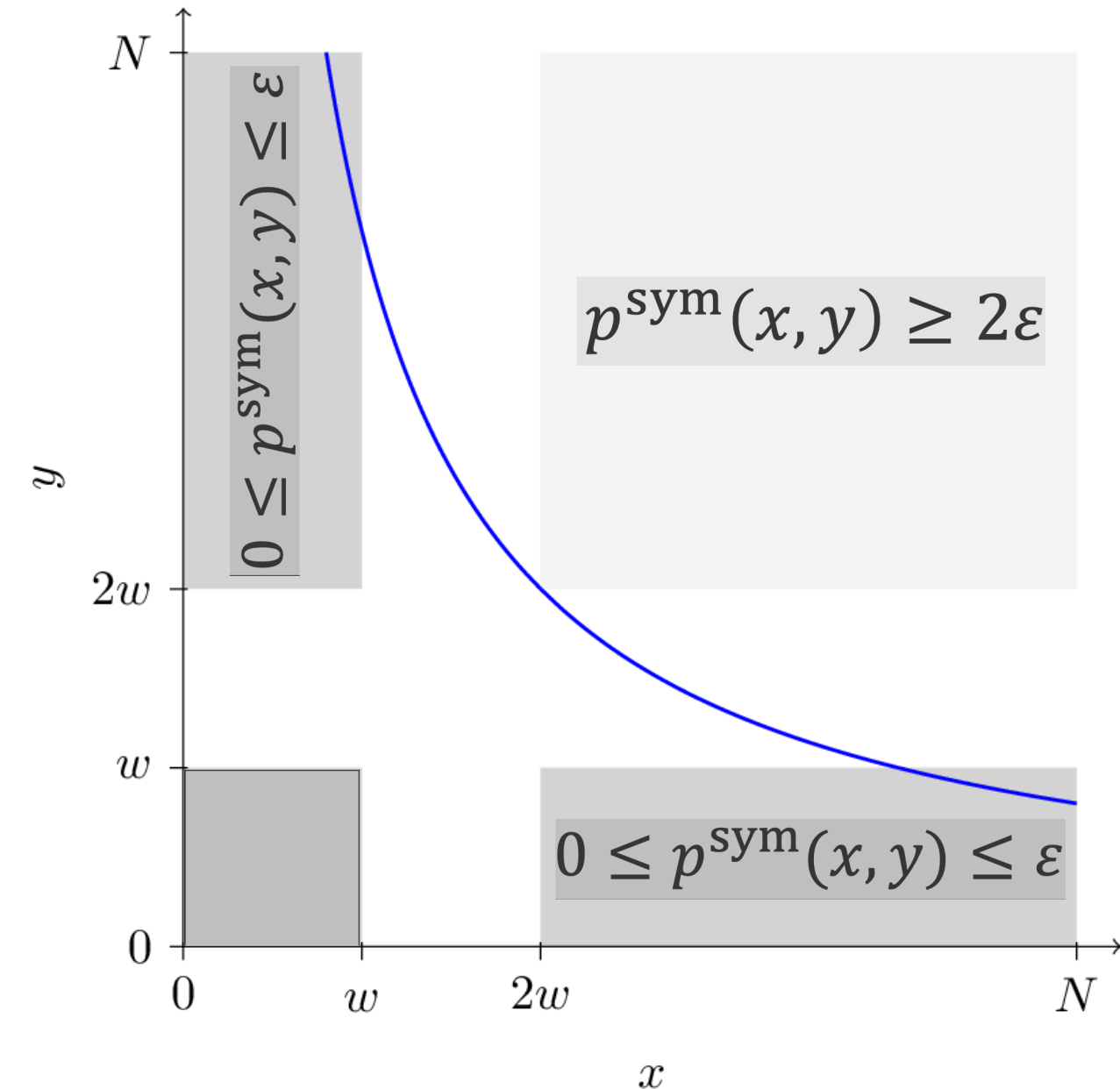
Note: WLOG  $p^{\text{sym}}(x, y) = p^{\text{sym}}(y, x)$ .

# Underlying Polynomial Question



- Must lower-bound  $\deg(p^{\text{sym}})$  where  $p^{\text{sym}}$  is as shown on the left.
- $p^{\text{sym}}$  is obtained by applying Marriott-Watrous transformation to a QMA protocol

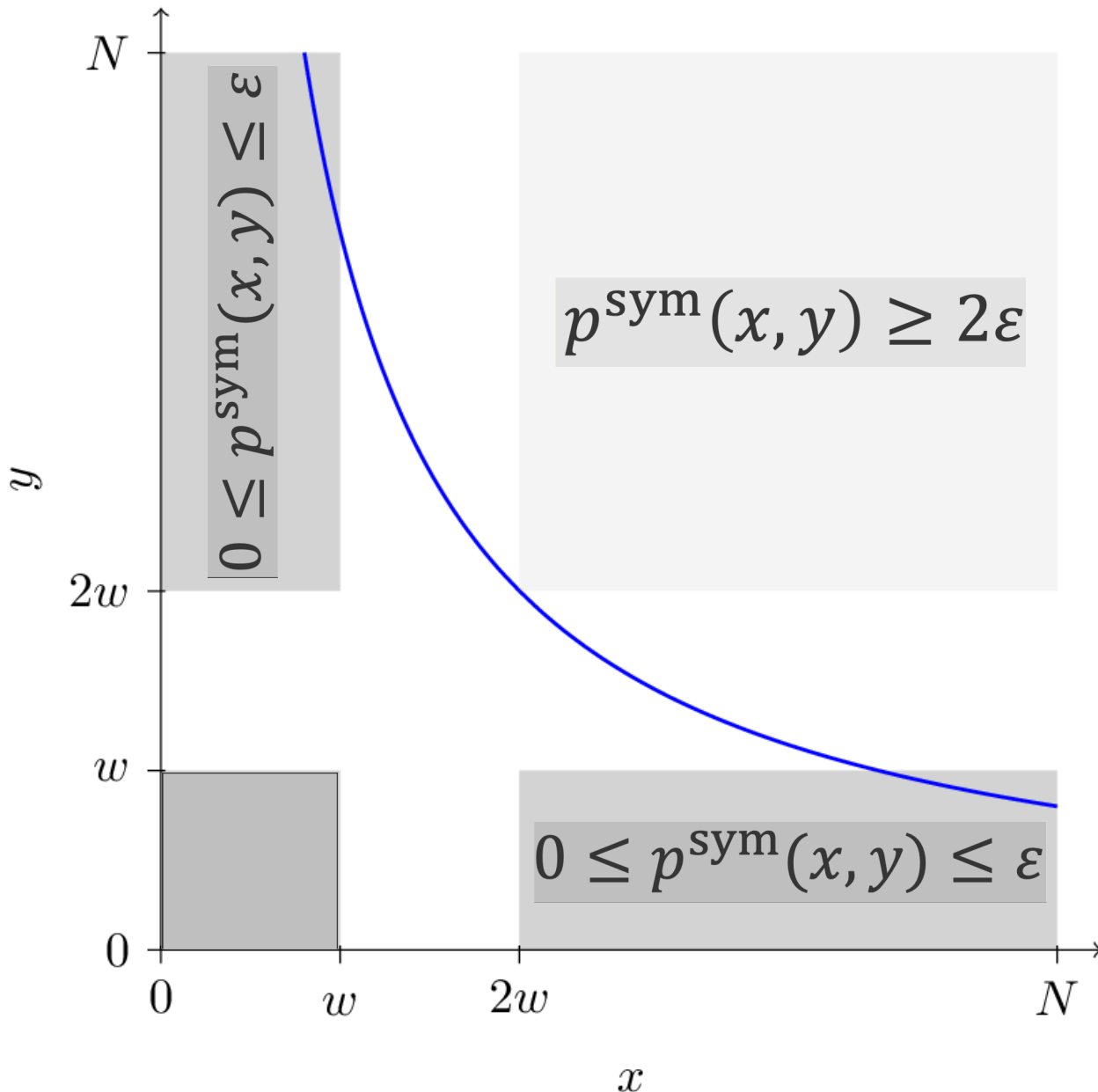
# Idea: Restrict $p^{\text{sym}}$ to a Hyperbola!



Let  $q(x) = \varepsilon^{-1} \cdot p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$ .

This is a univariate **Laurent** polynomial of degree and anti-degree  $\leq \deg(p)$ .

# Idea: Restrict $p^{\text{sym}}$ to a Hyperbola!



Let  $q(x) = \varepsilon^{-1} \cdot p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$ .

This is a univariate **Laurent** polynomial of degree and anti-degree  $\leq \deg(p)$ .

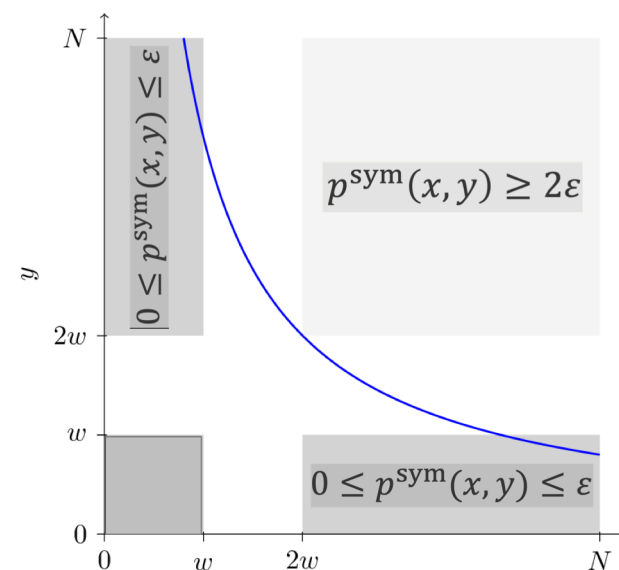
- $q(1) \geq 2$ .
- For any  $x \in [2, N/w]$ ,  $\left(2wx, \frac{2w}{x}\right)$  is in the bottom-right box, so it seems like  $|q(x)| \leq 1$ .
- **Problem:** We only have control of  $p^{\text{sym}}$ 's values at **integer** inputs, and hence  $q$ 's values only at inputs 1 and 2. Let's ignore for now.



# Summarizing Previous Slide

Let  $q(x) = \varepsilon^{-1} p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$ . This is a univariate **Laurent** polynomial in  $x$  of degree and anti-degree at most  $d := \deg(p)$ , such that:

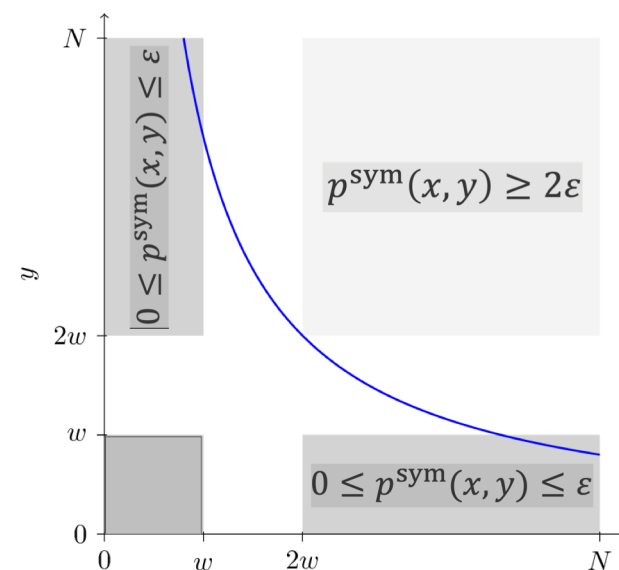
- $q(1) \geq 2$ .
- For any  $x \in [2, n/w]$ ,  $|q(x)| \leq 1$ .



# Summarizing Previous Slide

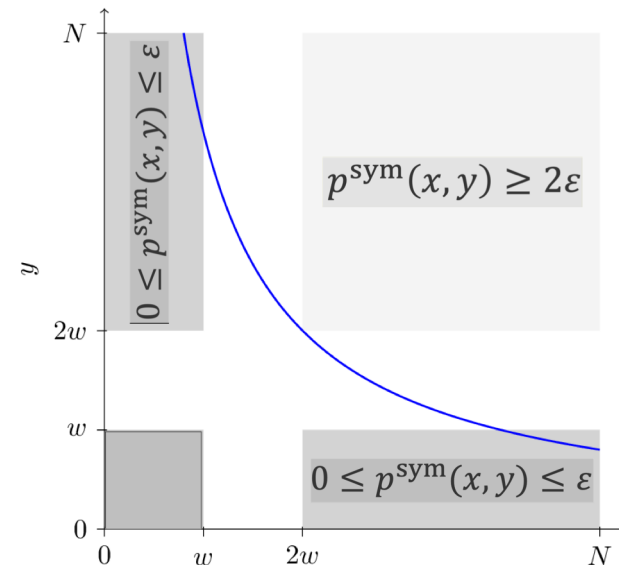
Let  $q(x) = \varepsilon^{-1} p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$ . This is a univariate **Laurent** polynomial in  $x$  of degree and anti-degree at most  $d := \deg(p)$ , such that:

- $q(1) \geq 2$ .
- For any  $x \in [2, n/w]$ ,  $|q(x)| \leq 1$ .
- If  $q$  were a **standard** polynomial of degree  $d$ , Markov's inequality would imply that  $d \geq \sqrt{n/w}$ .



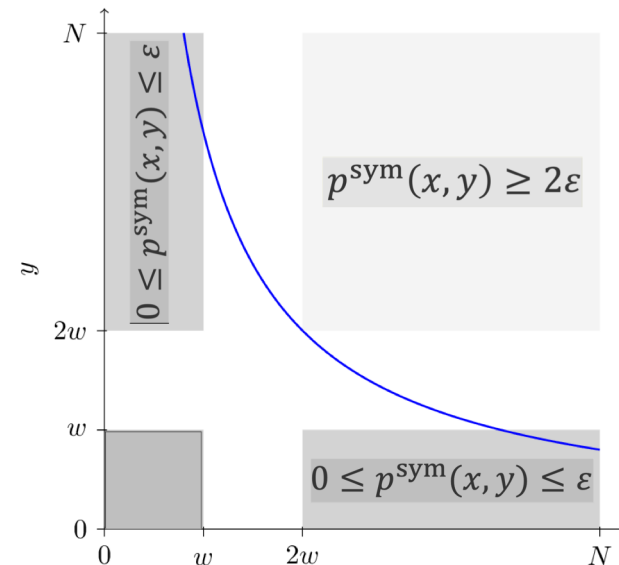
# Change of Variable

- **Next Key Lemma:**  $q(x) = \varepsilon^{-1} p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$  is actually a **standard** polynomial in  $(x + 1/x)$  of degree at most  $d$ .
- Proof:
  - Recall WLOG  $p^{\text{sym}}(|S|, |T|)$  is symmetric in its two inputs.
  - The fundamental theorem of symmetric polynomials says:  $p^{\text{sym}}$  is a degree  $d$  polynomial in the elementary symmetric polynomials:  $|S| + |T|$  and  $|S| \cdot |T|$ .
  - But  $q$  is the restriction of  $p^{\text{sym}}$  to a hyperbola  $\left(2wx, \frac{2w}{x}\right)$ .
    - On which  $|S| \cdot |T|$  is constant (i.e.,  $|S| \cdot |T| = 4w^2$ ).



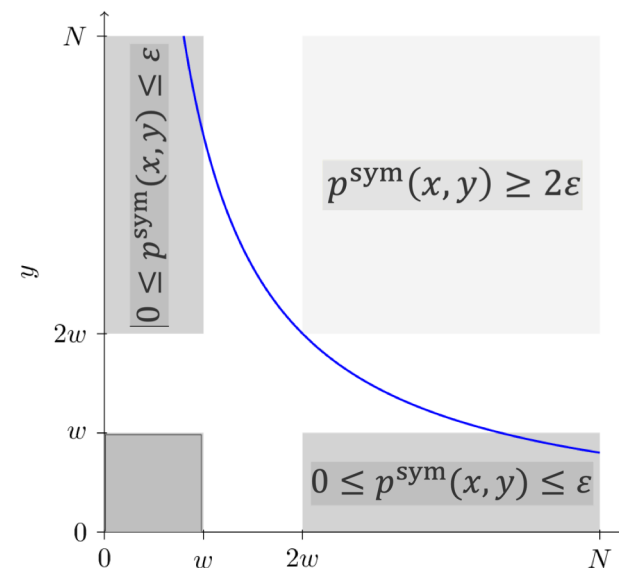
# Change of Variable

- **Next Key Lemma:**  $q(x) = \varepsilon^{-1} p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$  is actually a **standard** polynomial in  $(x + 1/x)$  of degree at most  $d$ .
- Proof:
  - Recall WLOG  $p^{\text{sym}}(|S|, |T|)$  is symmetric in its two inputs.
  - The fundamental theorem of symmetric polynomials says:  $p^{\text{sym}}$  is a degree  $d$  polynomial in the elementary symmetric polynomials:  $|S| + |T|$  and  $|S| \cdot |T|$ .
  - But  $q$  is the restriction of  $p^{\text{sym}}$  to a hyperbola  $\left(2wx, \frac{2w}{x}\right)$ .
    - On which  $|S| \cdot |T|$  is constant (i.e.,  $|S| \cdot |T| = 4w^2$ ).
  - So  $q$  is actually a degree  $d$  polynomial in  $|S| + |T|$ .
  - On the hyperbola,  $|S| + |T| = 2w(x + 1/x)$ .
  - So  $q$  is actually a degree  $d$  polynomial in  $(x + 1/x)$ .



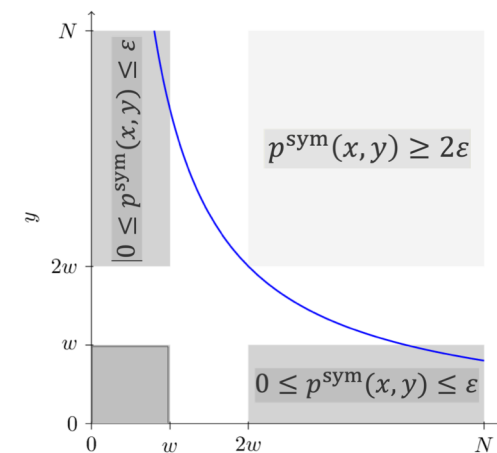
# Completing the Argument

- **Recall:**  $q(x) = \varepsilon^{-1} p^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$  is actually a **standard** polynomial in  $(x + 1/x)$  of degree at most  $d$ .
- Let  $t = x + 1/x$  and  $r(t) = q(x)$ . Then:
  - $\deg(r(t)) \leq d$
  - $r(2) = \varepsilon^{-1} p^{\text{sym}}(2w, 2w) \geq 2$
  - $|r(t)| \leq 1$  for all  $t \in \left[2.5, \frac{n}{w} + \frac{w}{n}\right]$ .
  - Markov's inequality implies that  $d \geq \sqrt{n/w}$ .



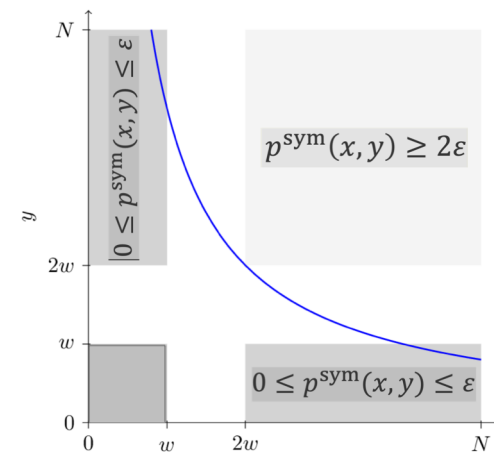
# Addressing the Ignored Issue

- **Problem:** We only have control of  $p^{\text{sym}}_s$  values at *integer* inputs, and hence  $q$ 's values only at inputs 1 and 2.



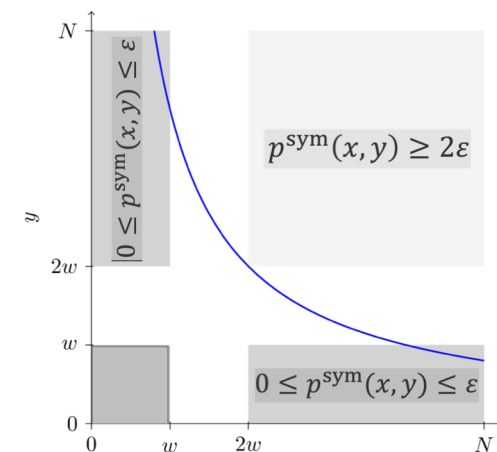
# Addressing the Ignored Issue

- **Problem:** We only have control of  $p^{\text{sym}}_S$  values at *integer* inputs, and hence  $q$ 's values only at inputs 1 and 2.
- Sketch of how to deal with this:
  - Recall that for integer inputs  $(x, y)$ ,  $p^{\text{sym}}(x, y) = \mathbf{E}_{|S|=x, |T|=y}[p(S, T)]$ .



# Addressing the Ignored Issue

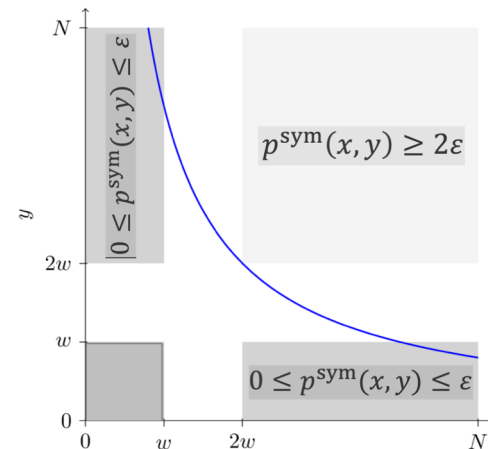
- **Problem:** We only have control of  $p^{\text{sym}}'$ 's values at *integer* inputs, and hence  $q$ 's values only at inputs 1 and 2.
- Sketch of how to deal with this:
  - Replace  $p^{\text{sym}}$  with a different symmetrization of  $p$  that is bounded even at non-integer inputs, namely:





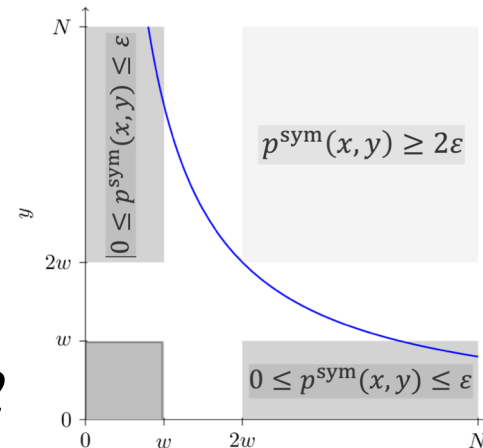
# Addressing the Ignored Issue

- **Problem:** We only have control of  $p^{\text{sym}}'$ 's values at *integer* inputs, and hence  $q$ 's values only at inputs 1 and 2.
- Sketch of how to deal with this:
  - Replace  $p^{\text{sym}}$  with a different symmetrization of  $p$  that is bounded even at non-integer inputs, namely:
    - $p_{\text{new}}^{\text{sym}}(x, y) = \mathbf{E}_{S, T}[p(S, T)]$  where each coordinate of  $S$  and  $T$  are drawn iid such that the **expected values** of  $|S|$  and  $|T|$  are  $x$  and  $y$ .
    - Since  $p$  is bounded at all Boolean inputs  $S, T$ ,  $p_{\text{new}}^{\text{sym}}(x, y)$  is bounded at all inputs in  $[0, n] \times [0, n]$  (even non-integers).



# Addressing the Ignored Issue

- **Problem:** We only have control of  $p^{\text{sym}'}$ 's values at **integer** inputs, and hence  $q$ 's values only at inputs 1 and 2.
- Sketch of how to deal with this:
  - Replace  $p^{\text{sym}}$  with a different symmetrization of  $p$  that is bounded even at non-integer inputs, namely:
    - $p_{\text{new}}^{\text{sym}}(x, y) = \mathbf{E}_{S, T}[p(S, T)]$  where each coordinate of  $S$  and  $T$  are drawn iid such that the **expected values** of  $|S|$  and  $|T|$  are  $x$  and  $y$ .
    - Since  $p$  is bounded at all Boolean inputs  $S, T$ ,  $p_{\text{new}}^{\text{sym}}(x, y)$  is bounded at all inputs in  $[0, n] \times [0, n]$  (even non-integers).
  - Introduces a new problem:
    - We now have **less** control over  $p_{\text{new}}^{\text{sym}'}$ 's behavior at **integer** inputs.
    - $q(x) := p_{\text{new}}^{\text{sym}}\left(2wx, \frac{2w}{x}\right)$  may not have a "jump" between  $x=1$  and  $x=2$



Second Result:  
Quantum Algorithms That Can  
Sample From  $S$

# Sampling from $S$

- In applications, when trying to estimate the size of a set  $S \subseteq [n]$ , often we can do more than make membership queries to  $S$ .
  - Often we can efficiently generate nearly uniform samples from  $S$  (e.g., via Markov Chain Monte Carlo).
    - If  $S$  is the set of perfect matchings in a bipartite graph [Jerrum, Sinclair, and Vigoda 2004].
    - Or the set of grid points in a high-dimensional convex body [Dyer, Frieze, and Kannan 1991].

# Sampling from $S$

- In applications, when trying to estimate the size of a set  $S \subseteq [n]$ , often we can do more than make membership queries to  $S$ .
- Question: If we can make membership queries to  $S$ , **and** sample uniformly from  $S$ , how efficiently can we solve  $AC_{w,n}$ ?

# Sampling from $S$

- In applications, when trying to estimate the size of a set  $S \subseteq [n]$ , often we can do more than make membership queries to  $S$ .
- Question: If we can make membership queries to  $S$ , **and** sample uniformly from  $S$ , how efficiently can we solve  $AC_{w,n}$ ?
- **CLASSICAL SOLUTIONS**
  - $O(n/w)$  classical membership queries to  $S$ 
    - Randomly pick universe elements and see if any are in  $S$
  - $O(\sqrt{w})$  classical samples from  $S$ 
    - Birthday Paradox: sample from  $S$  and see if any two samples are the same.

# Quantum Sampling from $S$

- Suppose the quantum algorithm is also given copies of the state:

$$|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{i \in S} |i\rangle$$

- Models situations where  $S$  can be efficiently “QSampled” (Aharonov & Ta-Shma 2003)

# Quantum Sampling from $S$

- Suppose the quantum algorithm is also given copies of the state:

$$|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{i \in S} |i\rangle$$

- Models situations where  $S$  can be efficiently “QSampled” (Aharonov & Ta-Shma 2003)
  - Many interesting sets can be efficiently QSampled, including perfect matchings [JSV04] and grid points in convex bodies [DFK91].
  - All problems in SZK can be efficiently reduced to some instance of QSampling.



# Quantum Sampling from $S$

- Suppose the quantum algorithm is also given copies of the state:

$$|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{i \in S} |i\rangle$$

- Models situations where  $S$  can be efficiently “QSampled” (Aharonov & Ta-Shma 2003)
- Then known quantum query lower bounds no longer apply.

# Quantum Sampling from $S$

- Suppose the quantum algorithm is also given copies of the state:

$$|S\rangle := \frac{1}{\sqrt{|S|}} \sum_{i \in S} |i\rangle$$

- Models situations where  $S$  can be efficiently “QSampled” (Aharonov & Ta-Shma 2003)
- Then known quantum query lower bounds no longer apply.
  - All the more so if the algorithm can also query an oracle that **reflects** about  $|S\rangle$ : i.e., can apply the unitary transformation  $U = I - 2|S\rangle\langle S|$ .
  - The ability to perform **reflect** about  $|S\rangle$  follows in a black-box way from the ability to prepare the state  $|S\rangle$  unitarily.

# Upper Bounds

Recall: We can decide whether  $|S| \leq w$  or  $|S| \geq 2w$  using:

- **CLASSICAL SOLUTIONS**

1.  $T = O(n/w)$  classical membership queries to  $S$
2.  $R = O(\sqrt{w})$  classical samples from  $S$

# Upper Bounds

Recall: We can decide whether  $|S| \leq w$  or  $|S| \geq 2w$  using:

- **CLASSICAL SOLUTIONS**

1.  $T = O(n/w)$  classical membership queries to  $S$
2.  $R = O(\sqrt{w})$  classical samples from  $S$

- **QUANTUM SOLUTIONS**

1.  $T = O(\sqrt{n/w})$  quantum membership queries to  $S$  (BHT 1998)

# Upper Bounds

Recall: We can decide whether  $|S| \leq w$  or  $|S| \geq 2w$  using:

- **CLASSICAL SOLUTIONS**

1.  $T = O(n/w)$  classical membership queries to  $S$
2.  $R = O(\sqrt{w})$  classical samples from  $S$

- **QUANTUM SOLUTIONS**

1.  $T = O(\sqrt{n/w})$  quantum membership queries to  $S$  (BHT 1998)
2.  $R = O(\min(\sqrt{n/w}, w^{1/3}))$  copies of  $|S\rangle$  and reflections
  - $O(\sqrt{n/w})$ : project  $|S\rangle$  onto  $|1\rangle + \dots + |N\rangle$  and do amplitude amplification
  - $O(w^{1/3})$ : Use “quantum collision” algorithm (BHT 1998) in a new way

# Our Result

**Theorem:** Given  $S \subseteq [n]$ , any quantum algorithm that solves  $AC_{w,n}$  using  $T$  queries to  $S$  as well as  $R$  copies of  $|S\rangle$  and reflections about  $|S\rangle$ , requires either:

$$T = \Omega(\sqrt{n/w}) \text{ or } R = \Omega(\min(\sqrt{n/w}, w^{1/3}))$$

# Proof of Lower Bound for Quantum Query+QSampling Algorithms for $AC_{w,n}$

# Recall Result 1

**Theorem:** Given  $S \subseteq [n]$ , any quantum algorithm to decide whether  $|S| \leq w$  or  $|S| \geq 2w$ , using  $T$  queries to  $S$  as well as  $R$  copies of  $|S\rangle$  and reflections about  $|S\rangle$ , requires either:

$$T = \Omega(\sqrt{n/w}) \text{ or } R = \Omega(\min(\sqrt{n/w}, w^{1/3}))$$



**Key Lemma:** Suppose a quantum algorithm gets  $R$  copies of  $|S\rangle$  and makes  $T$  membership queries to set  $S$  with indicator vector  $x$ .

Let  $q(k)$  be its acceptance probability, averaged over all  $S \subseteq [n]$ , with  $|S| = k$ . Then  $q(k)$  is a Laurent polynomial of degree  $\leq 2(T + R)$  and antidegree  $\leq R$ .

**Key Lemma:** Suppose a quantum algorithm gets  $R$  copies of  $|S\rangle$  and makes  $T$  membership queries to set  $S$  with indicator vector  $x$ .

Let  $q(k)$  be its acceptance probability, averaged over all  $S \subseteq [n]$ , with  $|S| = k$ . Then  $q(k)$  is a Laurent polynomial of degree  $\leq 2(T + R)$  and antidegree  $\leq R$ .

**Proof In Classical Case:** Consider an algorithm that takes  $R$  independent samples from  $S$ , and then (based on the sample) runs a classical decision tree of depth  $T$ .

**Key Lemma:** Suppose a quantum algorithm gets  $R$  copies of  $|S\rangle$  and makes  $T$  membership queries to set  $S$  with indicator vector  $x$ .

Let  $q(k)$  be its acceptance probability, averaged over all  $S \subseteq [n]$ , with  $|S| = k$ . Then  $q(k)$  is a Laurent polynomial of degree  $\leq 2(T + R)$  and antidegree  $\leq R$ .

**Proof In Classical Case:** Consider an algorithm that takes  $R$  independent samples from  $S$ , and then (based on the sample) runs a classical decision tree of depth  $T$ .

- The probability of getting ordered sample is  $\{i_1, \dots, i_R\}$  is  $\frac{1}{|S|^R} x_{i_1} \cdot \dots \cdot x_{i_R}$ .
- This is a degree- $R$  polynomial in  $x$ , weighted by  $\frac{1}{|S|^R}$ .

**Key Lemma:** Suppose a quantum algorithm gets  $R$  copies of  $|S\rangle$  and makes  $T$  membership queries to set  $S$  with indicator vector  $x$ .

Let  $q(k)$  be its acceptance probability, averaged over all  $S \subseteq [n]$ , with  $|S| = k$ . Then  $q(k)$  is a Laurent polynomial of degree  $\leq 2(T + R)$  and antidegree  $\leq R$ .

**Proof In Classical Case:** Consider an algorithm that takes  $R$  independent samples from  $S$ , and then (based on the sample) runs a classical decision tree of depth  $T$ .

- The probability of getting ordered sample is  $\{i_1, \dots, i_R\}$  is  $\frac{1}{|S|^R} x_{i_1} \cdot \dots \cdot x_{i_R}$ .
- This is a degree- $R$  polynomial in  $x$ , weighted by  $\frac{1}{|S|^R}$ .
- So probability of reaching any particular leaf is a degree- $(R + T)$  polynomial in  $x$ , weighted by  $\frac{1}{|S|^R}$ .

**Key Lemma:** Suppose a quantum algorithm gets  $R$  copies of  $|S\rangle$  and makes  $T$  membership queries to set  $S$  with indicator vector  $x$ .

Let  $q(k)$  be its acceptance probability, averaged over all  $S \subseteq [n]$ , with  $|S| = k$ . Then  $q(k)$  is a Laurent polynomial of degree  $\leq 2(T + R)$  and antidegree  $\leq R$ .

**Proof In Classical Case:** Consider an algorithm that takes  $R$  independent samples from  $S$ , and then (based on the sample) runs a classical decision tree of depth  $T$ .

- The probability of getting ordered sample is  $\{i_1, \dots, i_R\}$  is  $\frac{1}{|S|^R} x_{i_1} \cdot \dots \cdot x_{i_R}$ .
- This is a degree- $R$  polynomial in  $x$ , weighted by  $\frac{1}{|S|^R}$ .
- So probability of reaching any particular leaf is a degree- $(R + T)$  polynomial in  $x$ , weighted by  $\frac{1}{|S|^R}$ .
- Symmetrize this polynomial to get a degree- $(R + T)$  **univariate** polynomial in  $|S|$ , with weights proportional to  $\frac{1}{|S|^R}$ .
- This is a **Laurent** polynomial with the degree  $(R + T)$  and anti-degree  $R$ .

# Underlying Polynomial Question

Suppose  $p(k) = g(k) + h\left(\frac{1}{k}\right)$   $g, h$  univariate  
real polynomials

$$0 \leq p(k) \leq 1 \text{ for } k \in \{1, \dots, n\}$$

$$p(w) \leq \frac{1}{3}, \quad p(2w) \geq \frac{2}{3}$$

**Must Show:** Either

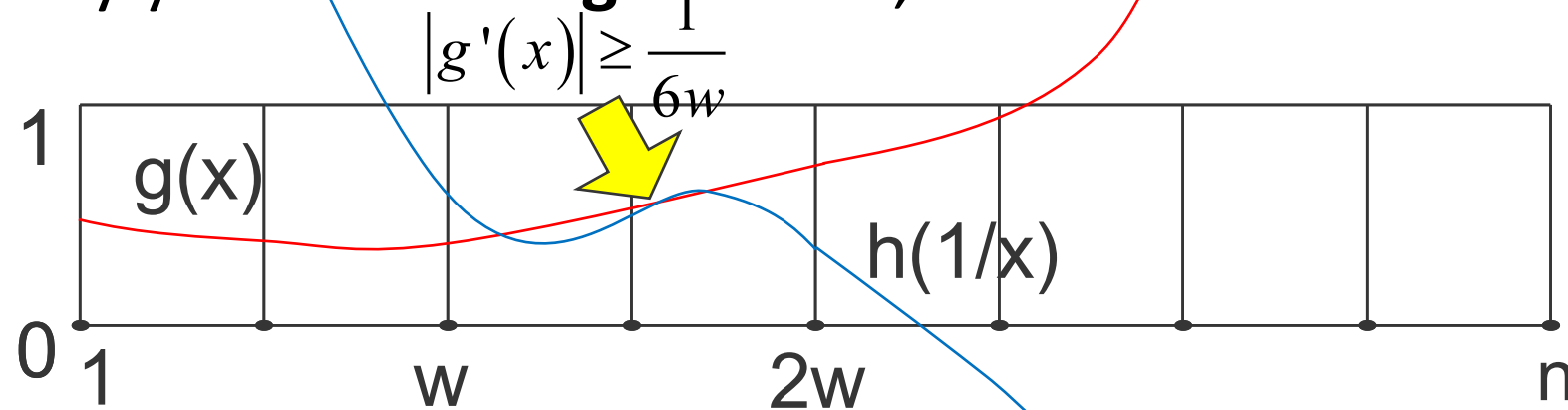
$$\deg(g) = \Omega\left(\sqrt{\frac{n}{w}}\right) \quad \text{or} \quad \deg(h) = \Omega(w^{1/4})$$

# “Explosion Argument”

- Either  $g$  or  $h$  must have a large derivative somewhere.
  - If it's low-degree, that means it takes large values (Markov).
  - But  $g(k) + h\left(\frac{1}{k}\right) \in [0,1]$  for all  $k \in \{1, \dots, n\}$ .
  - So the **other** polynomial must take large values of the opposite sign!
  - When switching from  $g$  to  $h$ , the x-axis gets compressed, so Markov's inequality yields even **larger** values, etc. etc.
- 
- But polynomials that grow without bound, on a compact set like  $[1, n]$  can never have existed in the first place

# “Explosion Argument”

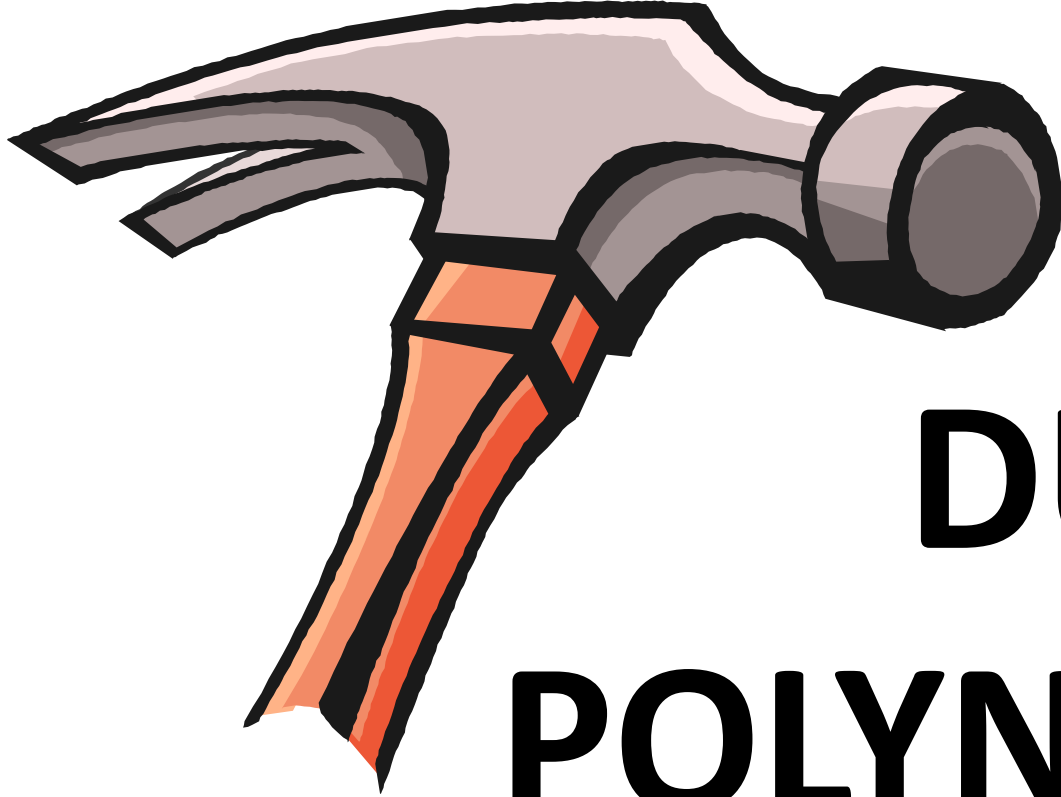
- Either  $g$  or  $h$  must have a large derivative somewhere.
- If it's low-degree, that means it takes large values (Markov).
- But  $g(k) + h\left(\frac{1}{k}\right) \in [0,1]$  for all  $k \in \{1, \dots, n\}$ .
- So the **other** polynomial must take large values of the opposite sign!
- When switching from  $g$  to  $h$ , the x-axis gets compressed, so Markov's inequality yields even **larger** values, etc. etc.



- But polynomials that grow without bound, on a compact set like  $[1, n]$  can never have existed in the first place



Tightening the  $\Omega(w^{1/4})$  to  $\Omega(w^{1/3})$



**DUAL  
POLYNOMIALS**

# Open Problems

- “Deep explanation” for why Laurent polynomials show up?
- Other applications of the Laurent polynomial method?
  - Kretschmer, recently: Simpler proof of  $\sim\sqrt{N}$  lower bound on approximate degree of AND-OR tree!
- Complexity of Approximate Counting with Queries+QSamples but **not** reflections?
- Lower-bound number of uses of a  $|0\rangle \leftrightarrow |S\rangle$  oracle?
- Is there a “real-world” (non-black-box) scenario where membership queries and QSampling are both easy, but approximate counting is hard?