Adversary Method

Adversary Method

The adversary method is a different technique to lower bound quantum query complexity.

The adversary method comes in two variations: the non-negative and negative weights versions.

The non-negative version is incomparable with the polynomial method.

The negative weights version characterizes quantum query complexity!

Adversary Method

The adversary method looks at the evolution of the state of an algorithm computing $f : S \to \{0, 1\}$ with $S \subseteq \{-1, +1\}^n$.

On input x, after t queries the algorithm is in the state

$$|\psi_x^t\rangle = U_t O_x \cdots U_1 O_x U_0 |0\rangle |0^m\rangle$$

We focus on how $\langle \psi_x^t | \psi_y^t \rangle$ changes over the course of the algorithm for pairs x, y with $f(x) \neq f(y)$.

Start of Algorithm

At the beginning of the algorithm, the state is independent of the input x.

$$|\psi_x^0\rangle = U_0|0\rangle|0^m\rangle$$

This means

$$\langle\psi_x^0|\psi_y^0\rangle = 1 \quad \text{for all} \quad x, y \in S\,.$$

End of Algorithm

At the end of the algorithm we make a measurement $\{\Pi_0, \Pi_1\}$.

If the algorithm has error probability $\leq \varepsilon$ then

$$\|\Pi_1|\psi_x^T\rangle\|^2 \leq \varepsilon \quad \text{for all} \quad x \in f^{-1}(0)$$

$$\|\Pi_0|\psi_y^T\rangle\|^2 \leq \varepsilon \quad \text{for all} \quad y \in f^{-1}(1)$$

Thus $|\langle\psi_x^T|\psi_y^T\rangle| = |\langle\psi_x^T|\Pi_0 + \Pi_1|\psi_y^T\rangle|$ $f(x) = 0, f(y) = 1$

$$\leq |\langle\psi_x^T|\Pi_0|\psi_y^T\rangle| + |\langle\psi_x^T|\Pi_1|\psi_y^T\rangle|$$

$$\leq 2\sqrt{\varepsilon} \qquad |\langle u|v\rangle| \leq \|u\|\|v\|$$

End of Algorithm

At the end of the algorithm we make a measurement $\{\Pi_0, \Pi_1\}$.

$f(x) = 0, f(y) = 1$

Thus $|\langle \psi_x^T | \psi_y^T \rangle| = |\langle \psi_x^T | \Pi_0 + \Pi_1 | \psi_y^T \rangle|$

$$\leq |\langle \psi_x^T | \Pi_0 | \psi_y^T \rangle| + |\langle \psi_x^T | \Pi_1 | \psi_y^T \rangle|$$

$$\leq 2\sqrt{\varepsilon}$$

Being a bit more careful you can show

$$|\langle \psi_x^T | \psi_y^T \rangle| \leq 2\sqrt{\varepsilon(1-\varepsilon)}$$

Change with a query

Next we look at the change in inner product with a query

$$|\psi_x^{t+1}\rangle = U_{t+1}O_x|\psi_x^t\rangle$$

$$|\psi_y^{t+1}\rangle = U_{t+1}O_y|\psi_y^t\rangle$$

$$\langle\psi_x^{t+1}|\psi_y^{t+1}\rangle = \langle\psi_x^t|O_x^*U_{t+1}^*U_{t+1}O_y|\psi_y^t\rangle$$

$$= \langle\psi_x^t|O_x^*O_y|\psi_y^t\rangle$$

Let $P_i = \sum_{z\in\{0,1\}^m} |i\rangle|z\rangle\langle z|\langle i|$ be the projector onto the subspace querying i.

Change with a query

Let $P_i = \sum_{z \in \{0,1\}^m} |i\rangle|z\rangle\langle z|\langle i|$ be the projector onto the subspace querying i.

Notice that $P_i P_j = \mathbf{0}$ and $\sum_{i=1}^{n} P_i = \mathbb{I}$.

$$O_x|\psi_x^t\rangle = \sum_{i=1}^{n} O_x P_i |\psi_x^t\rangle$$

$$= \sum_{i=1}^{n} x_i P_i |\psi_x^t\rangle$$

Change with a query

$$O_x|\psi_x^t\rangle = \sum_{i=1}^{n} x_i P_i |\psi_x^t\rangle$$

Thus

$$\langle\psi_x^t|O_x^* O_y|\psi_y^t\rangle = \sum_{i,j} x_i y_j \langle\psi_x^t|P_i P_j|\psi_y^t\rangle$$

$$= \sum_{i=1}^{n} x_i y_i \langle\psi_x^t|P_i|\psi_y^t\rangle$$

This means

$$\langle\psi_x^{t+1}|\psi_y^{t+1}\rangle - \langle\psi_x^t|P_i|\psi_y^t\rangle = \sum_{i=1}^{n} \langle\psi_x^t|P_i|\psi_y^t\rangle(x_i y_i - 1)$$

$$= -2 \sum_{i:x_i \neq y_i} \langle\psi_x^t|P_i|\psi_y^t\rangle$$

Adversary Method

Summing up, for any x, y with $f(x) \neq f(y)$

- Initially, $\langle \psi_x^0 | \psi_y^0 \rangle = 1$

- If the algorithm has error $\leq \varepsilon$ and makes T queries

$$|\langle \psi_x^T | \psi_y^T \rangle| \leq 2\sqrt{\varepsilon(1-\varepsilon)}$$

- With a single query

$$\langle \psi_x^{t+1} | \psi_y^{t+1} \rangle - \langle \psi_x^t | P_i | \psi_y^t \rangle = -2 \sum_{i : x_i \neq y_i} \langle \psi_x^t | P_i | \psi_y^t \rangle$$

Application

Intuition: It is easy to distinguish any single pair x, y.

But if we have to distinguish many pairs, and a single query only distinguishes a few of them, then we have to make many queries.

Example: Let's take $\mathrm{OR}_n : \{0,1\}^n \to \{0,1\}$.

$$X = \{0^n\} \qquad Y = \{10^{n-1}, 010^{n-2}, \ldots, 0^{n-1}1\}$$

I'll let $e_i = 0^{i-1}10^{n-i}$ so $Y = \{e_1, \ldots, e_n\}$.

Application to OR

We will show a lower bound for $\mathrm{OR}_n : \{0,1\}^n \to \{0,1\}$.

Define a potential function

$$W(t) = \sum_{i=1}^{n} \langle \psi_{0^n}^t | \psi_{e_i}^t \rangle$$

Then $W(0) = n$.

If the algorithm makes T queries and has error ε then $W(T) \leq 2n\sqrt{\varepsilon(1-\varepsilon)}$.

$$W(t) = \sum_{i=1}^{n} \langle \psi_{0^n}^t | \psi_{e_i}^t \rangle$$

$$W(t) - W(t+1) = \sum_{i=1}^{n} \langle \psi_{0^n}^t | \psi_{e_i}^t \rangle - \langle \psi_{0^n}^{t+1} | \psi_{e_i}^{t+1} \rangle$$

$$= 2 \sum_{i=1}^{n} \langle \psi_{0^n}^t | P_i | \psi_{e_i}^t \rangle$$

$$\leq 2 \sum_{i=1}^{n} \| P_i | \psi_{0^n}^t \rangle \|$$

$$\leq 2\sqrt{n}$$

Cauchy-Schwarz

$$\sum_{i=1}^{n} \| P_i | \psi_{0^n}^t \rangle \|^2 = 1$$

Putting it together

$$\left(1 - 2\sqrt{\varepsilon(1-\varepsilon)}\right) n \leq W(0) - W(T)$$

$$= (W(0) - W(1)) + \cdots + (W(T-1) - W(T))$$

$$\leq 2T\sqrt{n}$$

Putting it together gives

$$T \geq \frac{1 - 2\sqrt{\varepsilon(1-\varepsilon)}}{2}\sqrt{n}$$

In particular, $Q_{1/3}(\mathrm{OR}_n) = \Omega(\sqrt{n})$.

Adding weights

To get the most out of the bound, we can weight x, y pairs.

Let Γ be a $|f^{-1}(0)|$-by-$|f^{-1}(1)|$ non-negative matrix.

Intuition: we want to give larger weight to pairs x, y that are harder to distinguish.

Let u, v be such that $u^T \Gamma v = \|\Gamma\|$.

u, v non-negative

Define a potential function

$$W(t) = \sum_{\substack{x \in f^{-1}(0) \\ y \in f^{-1}(1)}} \Gamma(x,y) u(x) v(y) \langle \psi_x^t | \psi_y^t \rangle$$

Start of Algorithm

Define a potential function

$$W(t) = \sum_{\substack{x \in f^{-1}(0) \\ y \in f^{-1}(1)}} \Gamma(x,y) u(x) v(y) \langle \psi_x^t | \psi_y^t \rangle$$

Initially, $W(0) = \|\Gamma\|$.

End of Algorithm

After T queries

$$W(T) = \sum_{\substack{x \in f^{-1}(0) \\ y \in f^{-1}(1)}} \Gamma(x,y) u(x) v(y) \langle \psi_x^T | \psi_y^T \rangle$$

Γ non-negative

$$\leq \sum_{\substack{x \in f^{-1}(0) \\ y \in f^{-1}(1)}} \Gamma(x,y) u(x) v(y) |\langle \psi_x^T | \psi_y^T \rangle|$$

$$\leq 2\sqrt{\varepsilon(1-\varepsilon)} \sum_{\substack{x \in f^{-1}(0) \\ y \in f^{-1}(1)}} \Gamma(x,y) u(x) v(y)$$

$$\leq 2\sqrt{\varepsilon(1-\varepsilon)} \|\Gamma\|$$

Change with a query

For $i = 1, \ldots, n$ let D_i be a $|f^{-1}(0)|$ -by- $|f^{-1}(1)|$ matrix

$$D_i(x, y) = \begin{cases} 1 & \text{if } x_i \neq y_i \\ 0 & \text{otherwise} \end{cases}$$

You can show $|W(t+1) - W(t)| \leq 2 \max_{i=1,\ldots,n} \|\Gamma \circ D_i\|$.

Here $A \circ B$ is the entrywise product

$$(A \circ B)(x, y) = A(x, y)B(x, y)$$

Putting it all together

Define $$\mathrm{ADV}(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_{i=1,\ldots,n} \|\Gamma \circ D_i\|}$$

We have shown $$2T \max_{i=1,\ldots,n} \|\Gamma \circ D_i\| \geq \left(1 - 2\sqrt{\varepsilon(1-\varepsilon)}\right) \|\Gamma\|$$

This implies

$$Q_\varepsilon(f) \geq \frac{1 - 2\sqrt{\varepsilon(1-\varepsilon)}}{2} \mathrm{ADV}(f)$$

This is the (non-negative weight) adversary bound.

Negative weights

It is slightly unintuitive, but negative weights can help in the adversary bound. Now let Γ be arbitrary.

Initial: $W(0) = \|\Gamma\|$.

Change: $|W(t+1) - W(t)| \leq 2 \max_{i=1,\ldots,n} \|\Gamma \circ D_i\|$ (same proof).

Final: $W(T) \leq 2\sqrt{\varepsilon(1-\varepsilon)}\|\Gamma\|$ (proof much more subtle).

$$\mathrm{ADV}^{\pm}(f) = \max_{\Gamma} \min_{i=1,\ldots,n} \frac{\|\Gamma\|}{\|\Gamma \circ D_i\|}$$

$$Q_{\varepsilon}(f) \geq \frac{1 - 2\sqrt{\varepsilon(1-\varepsilon)}}{2} \mathrm{ADV}^{\pm}(f)$$

Negative weights

With negative weights the bound can be much larger.

With negative weights the adversary bound characterizes bounded-error quantum query complexity!

$$Q_{1/3}(f) = \Theta\left(\mathrm{ADV}^{\pm}(f)\right)$$

You can actually give algorithmic results by upper bounding the quantum adversary bound!