

Lower bounds on quantum query complexity

Part II

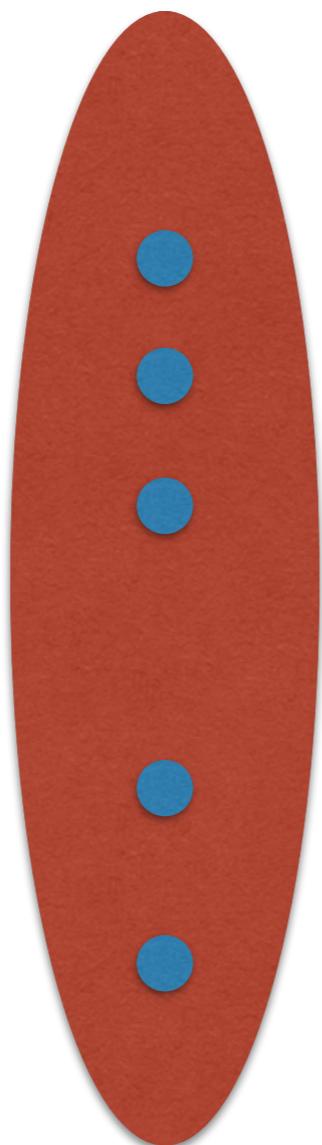
Marc Kaplan
Telecom ParisTech - University of Edinburgh

Adversary method(s)

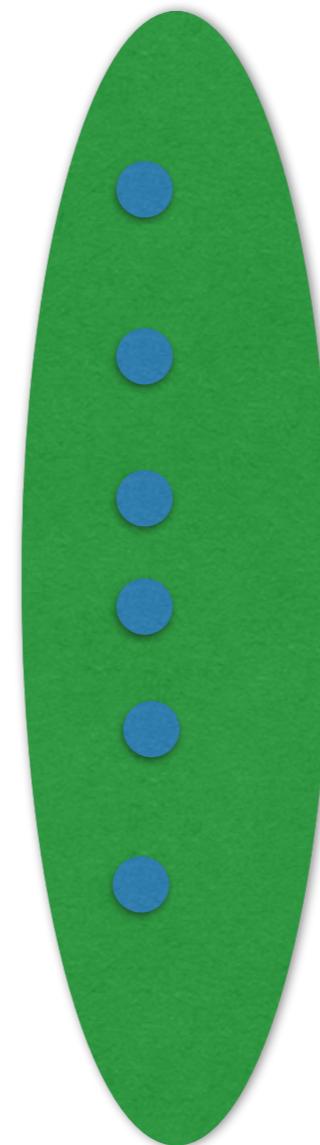
- Original adversary method
- Weighted adversary method
- Spectral adversary method
- Generalized adversary method

Adversary method: original version

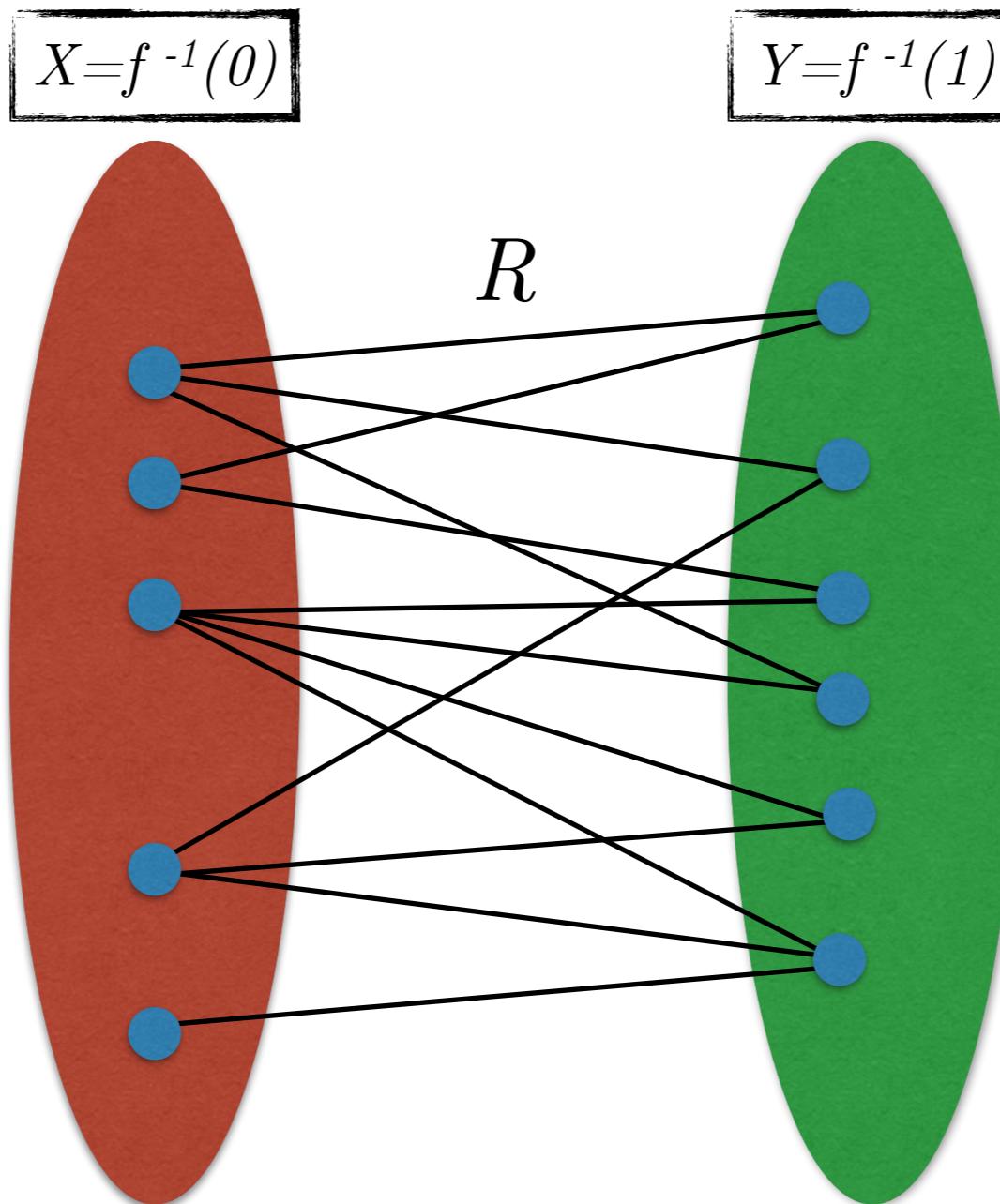
$$X=f^{-1}(0)$$



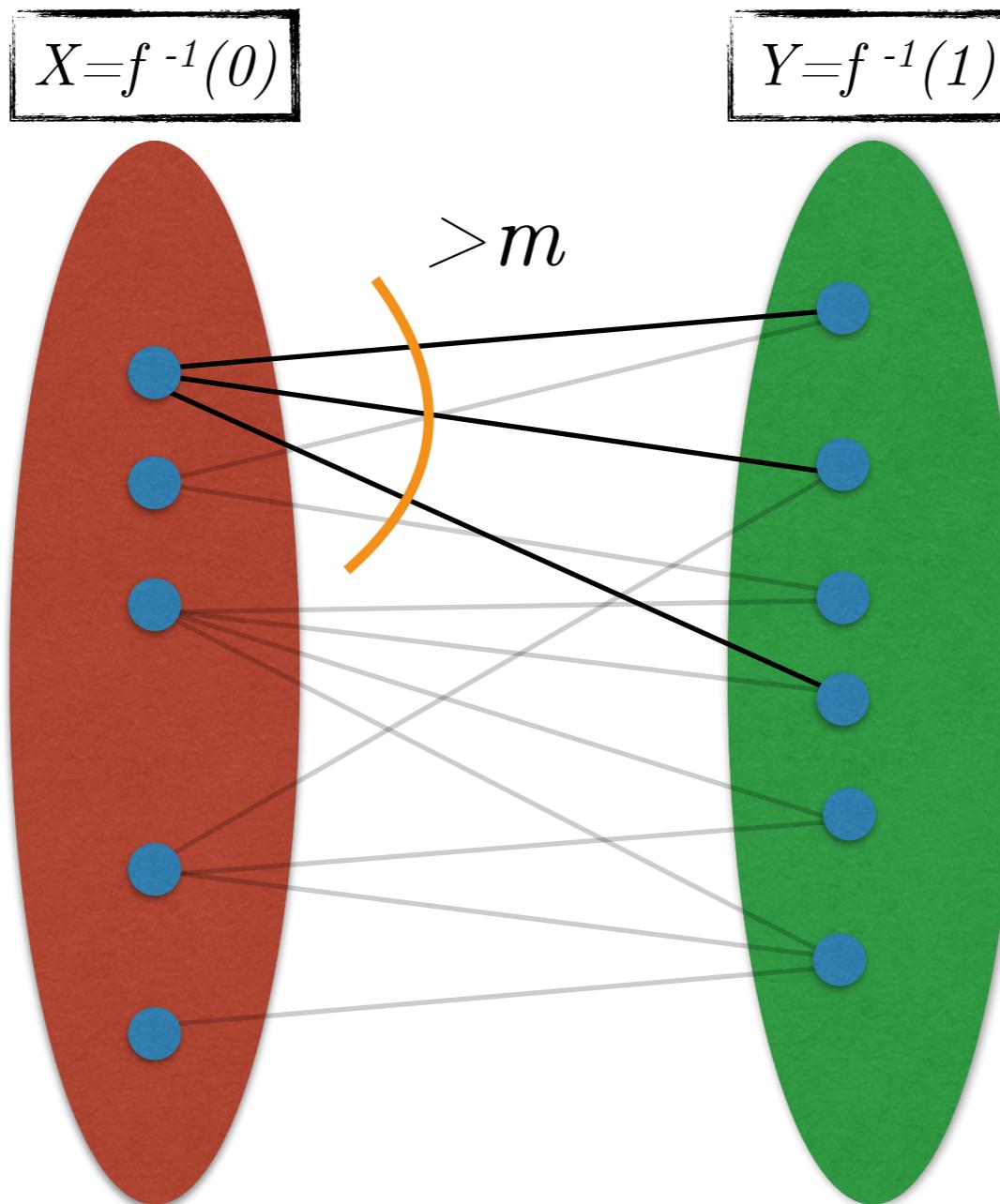
$$Y=f^{-1}(1)$$



Adversary method: original version

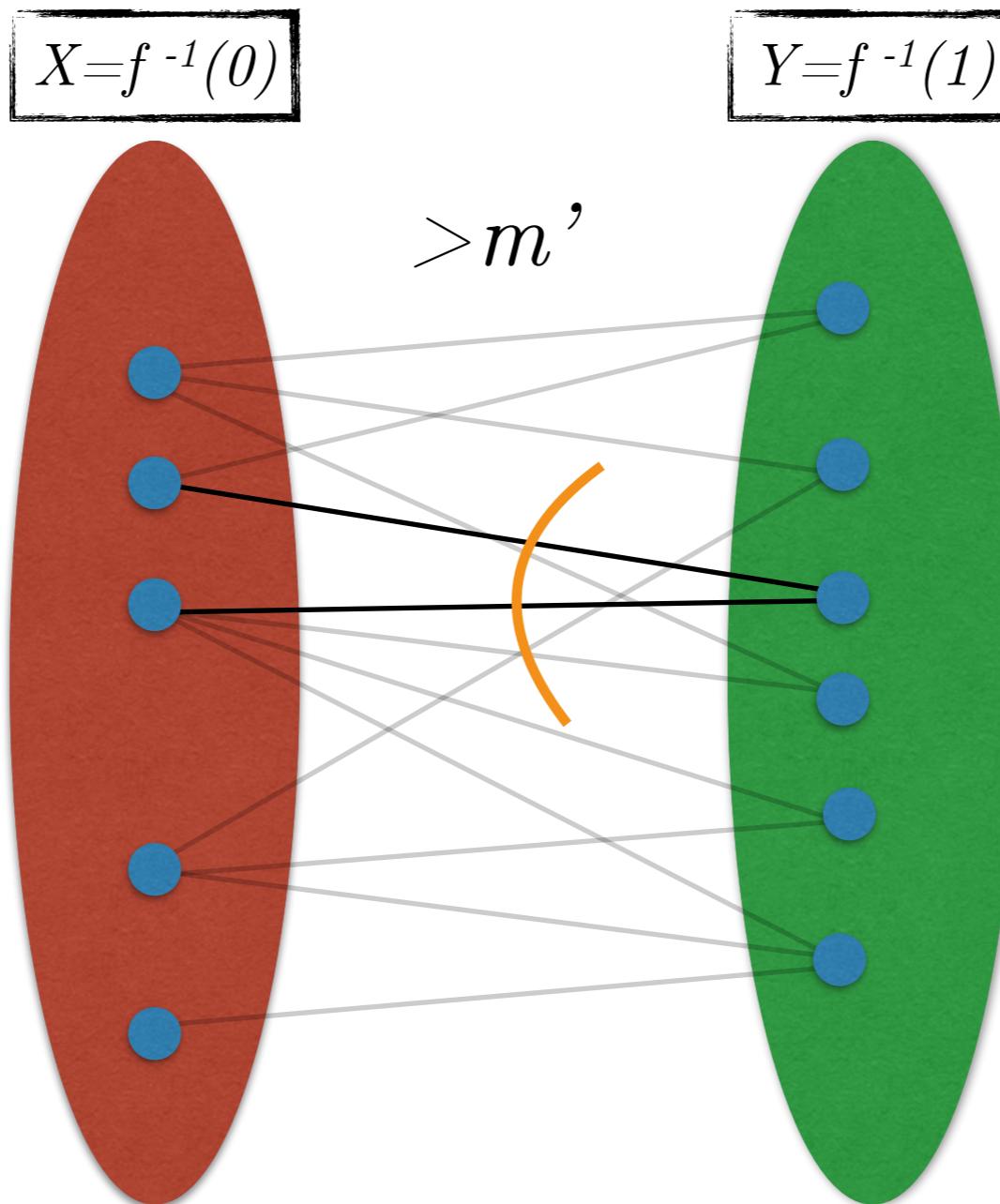


Adversary method: original version



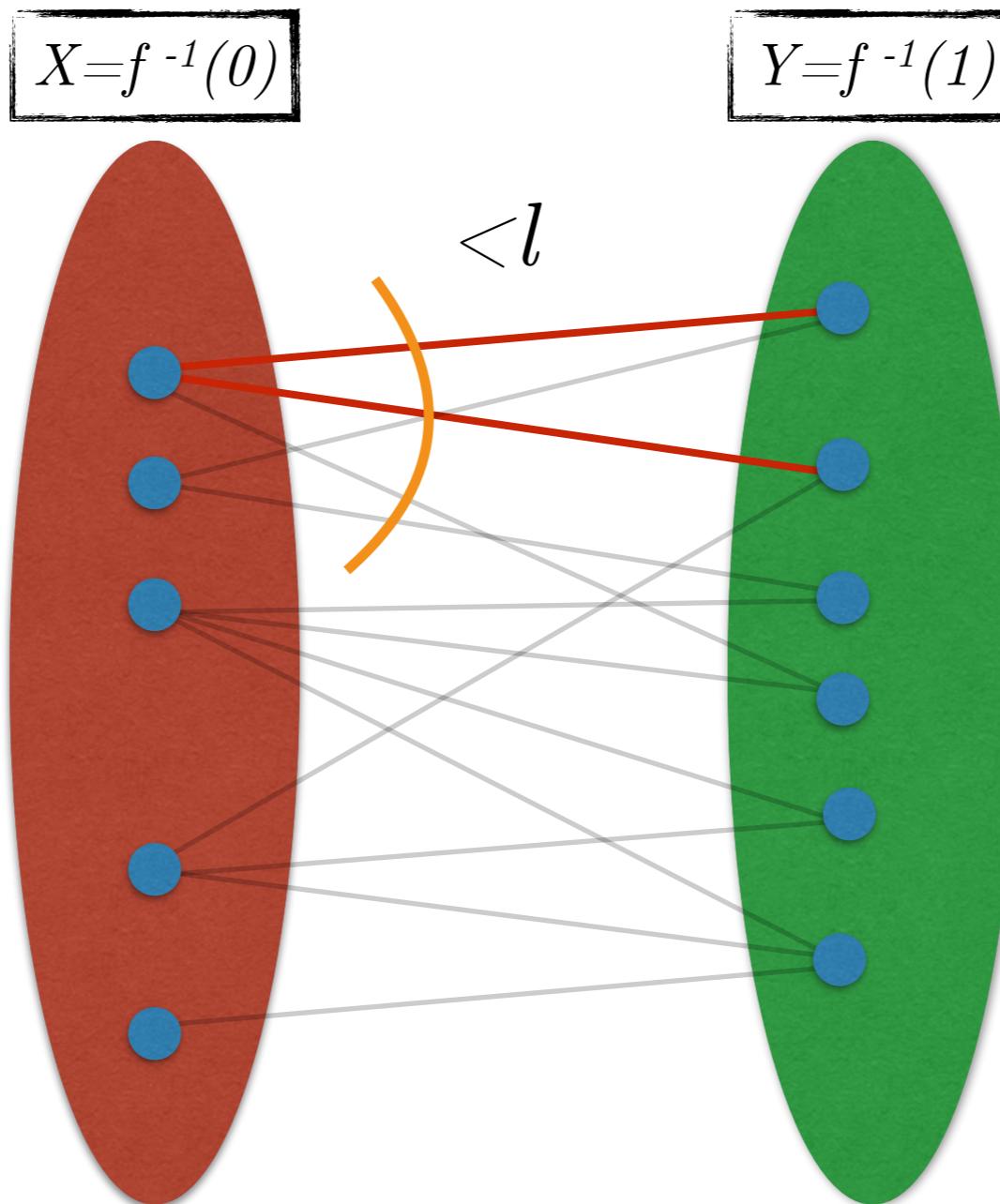
$$(x, y) \in R$$

Adversary method: original version



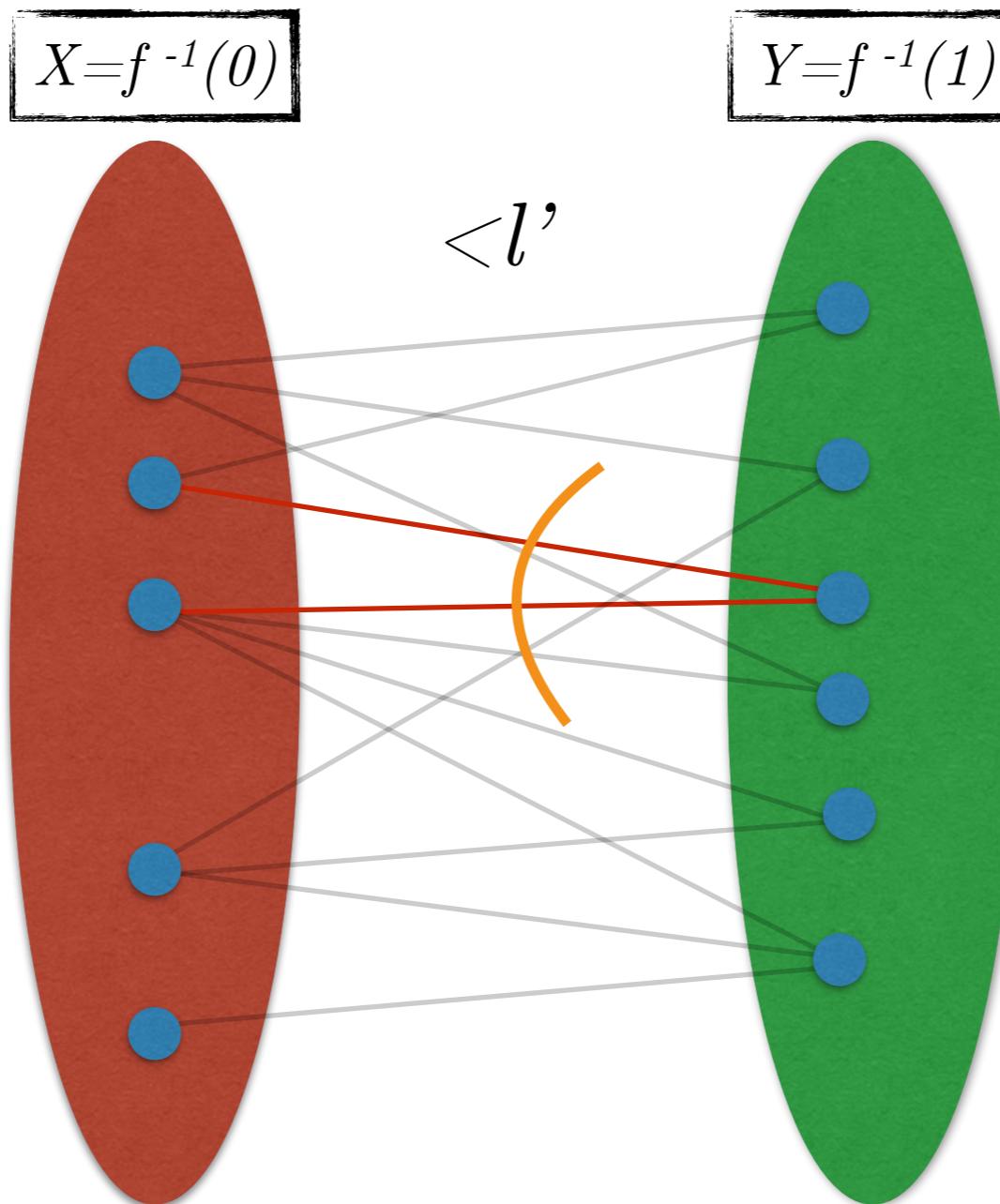
$$(x, y) \in R$$

Adversary method: original version



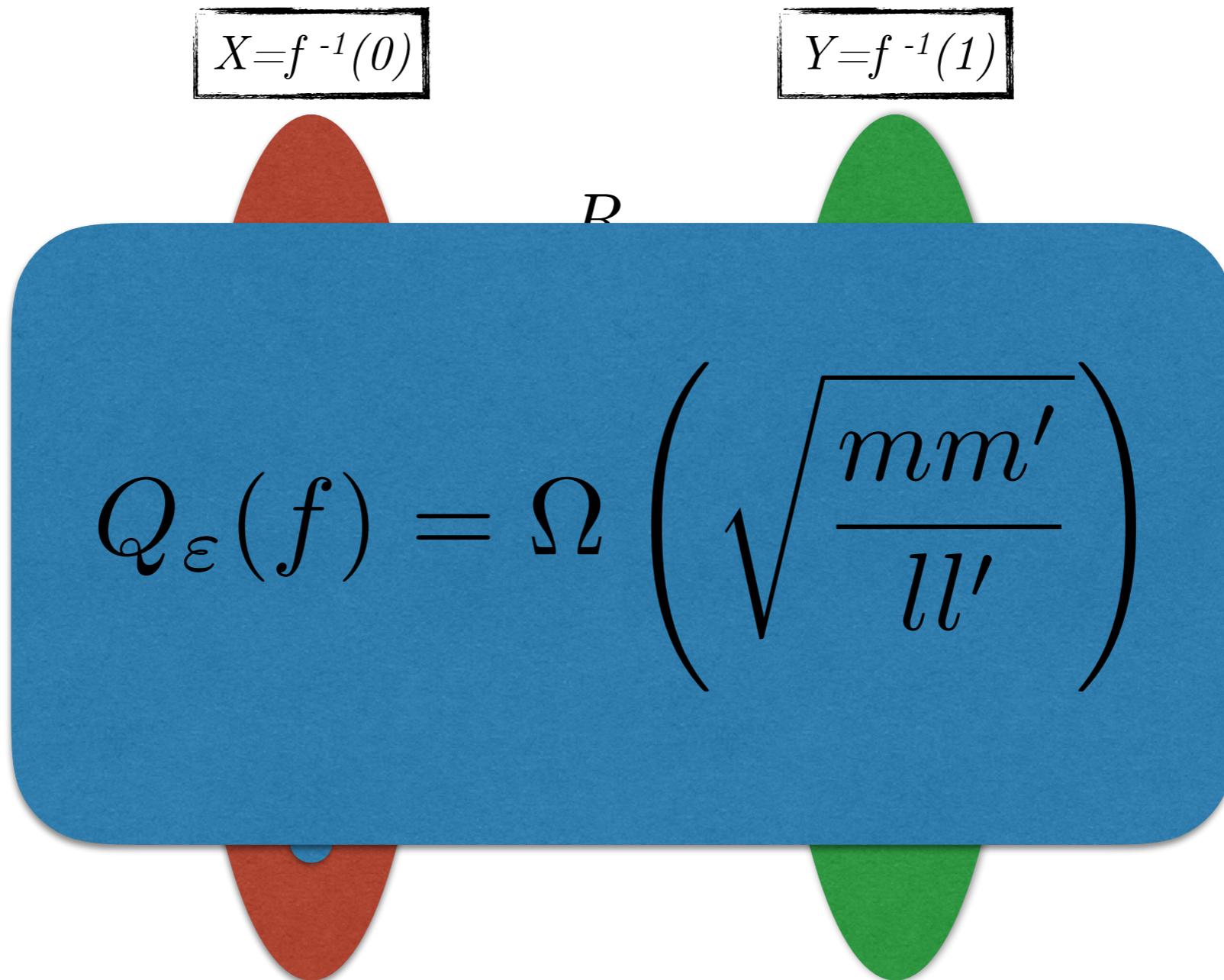
$(x, y) \in R$ and $x_i \neq y_i$

Adversary method: original version

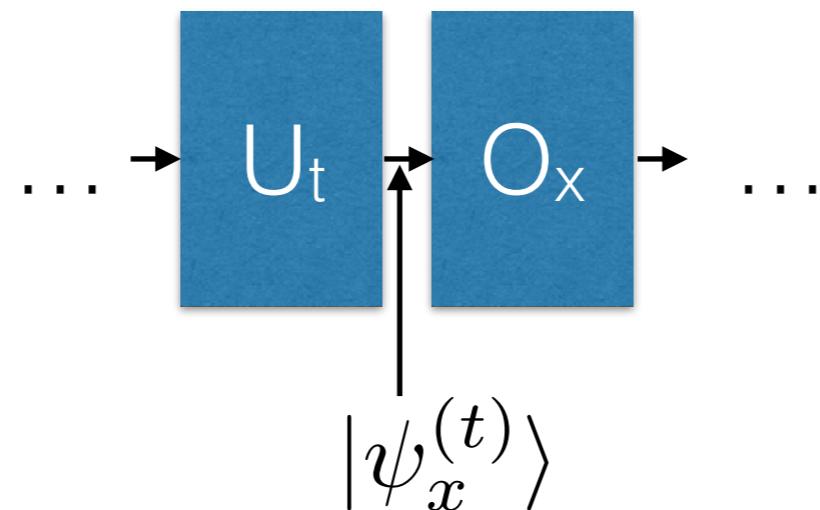


$(x, y) \in R$ and $x_i \neq y_i$

Adversary method: original version



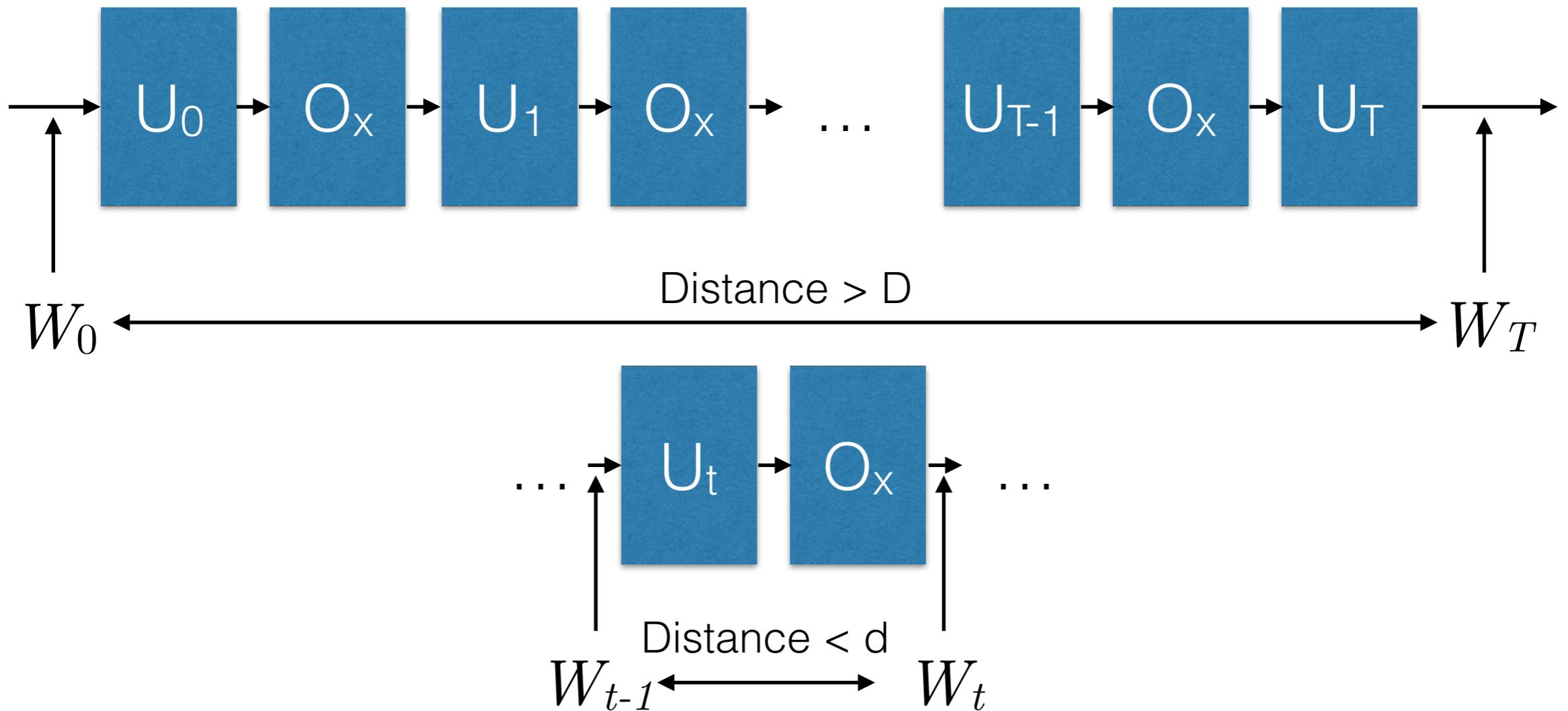
Proof sketch



Progress function:

$$W_t = \sum_{x,y \in R} \langle \psi_x^{(t)} | \psi_y^{(t)} \rangle$$

Proof sketch



1. $|W_T - W_0| = \Omega(\sqrt{|X||Y|m'm'})$
2. $|W_t - W_{t-1}| = O(\sqrt{|X||Y|ll'})$

Proof sketch - Part 1

$$\begin{aligned} W_0 &= \sum_{x,y \in R} \langle \Psi_x^{(0)} | \Psi_y^{(0)} \rangle \\ &= \# \{(x, y) \in R\} \\ &\geq \max \{|X|m, |Y|m'\} \\ &\geq \sqrt{|X||Y|mm'} \end{aligned}$$

$$W_T \simeq 0$$

Proof sketch - Part 2

$$\begin{aligned}|W_t - W_{t-1}| &\leq \sum_{x,y \in R} |\langle \psi_x^{(t)} | \psi_y^{(t)} \rangle - \langle \psi_x^{(t-1)} | \psi_y^{(t-1)} \rangle| \\&\leq \sum_{x,y \in R} |\langle O_x U_t \psi_x | O_y U_t \psi_y \rangle - \langle \psi_x | \psi_y \rangle| \\&\leq \max_i \sum_{x,y \in R} |\langle O_{x,i} U_t \psi_x | O_{y,i} U_t \psi_y \rangle - \langle \psi_x | \psi_y \rangle|\end{aligned}$$

Proof sketch - Part 2

$$\begin{aligned}|W_t - W_{t-1}| &\leq \sum_{x,y \in R} |\langle \psi_x^{(t)} | \psi_y^{(t)} \rangle - \langle \psi_x^{(t-1)} | \psi_y^{(t-1)} \rangle| \\&\leq \sum_{x,y \in R} |\langle O_x U_t \psi_x | O_y U_t \psi_y \rangle - \langle \psi_x | \psi_y \rangle| \\&\leq \max_i \sum_{x,y \in R} |\langle O_{x,i} U_t \psi_x | O_{y,i} U_t \psi_y \rangle - \langle \psi_x | \psi_y \rangle|\end{aligned}$$

Same unitaries
if $x_i = y_i$

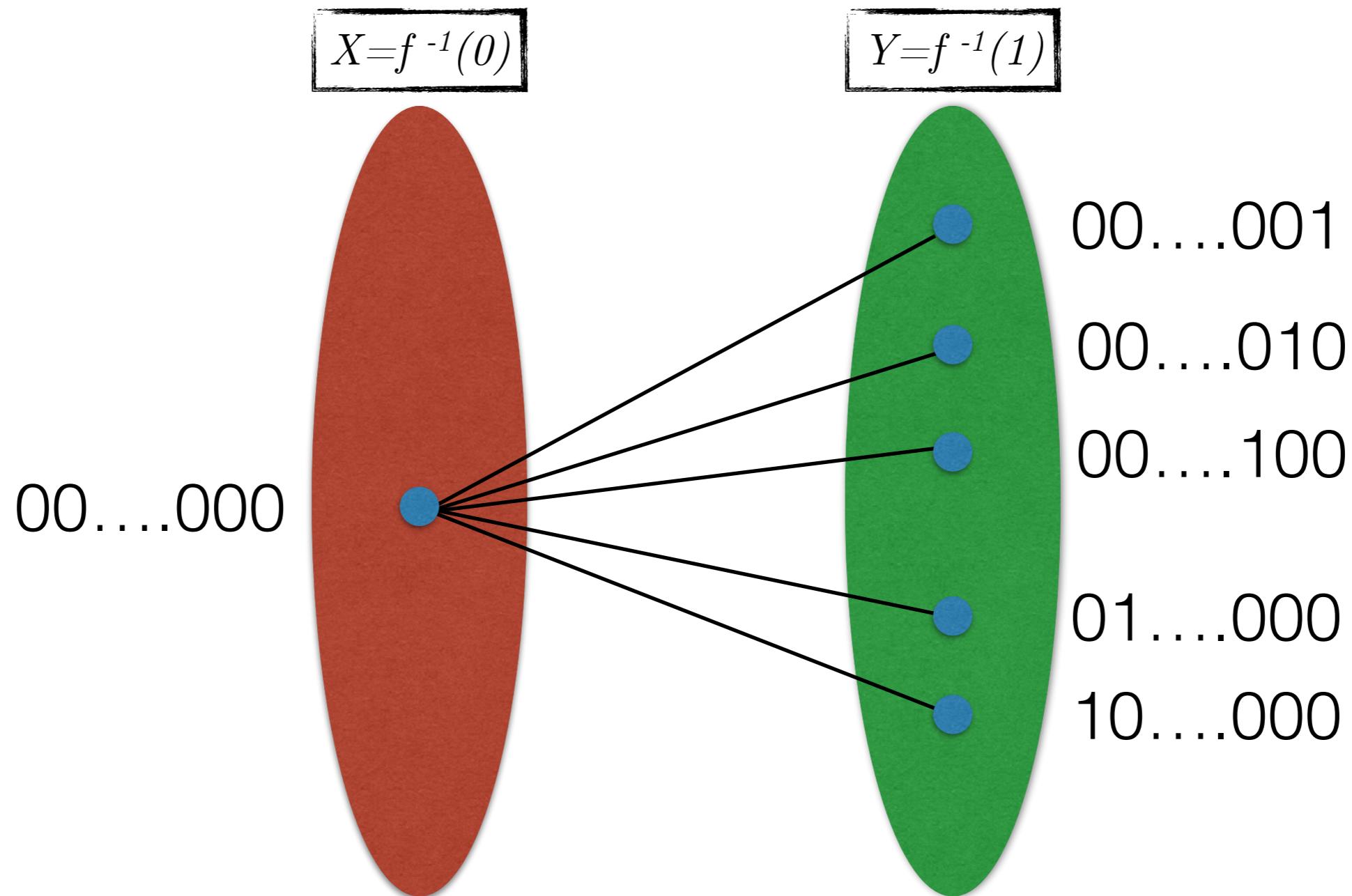
Proof sketch - Part 2

$$|W_t - W_{t-1}| \leq \max_i \sum_{x,y \in R} |\langle O_{x,i} U_t \psi_x | O_{y,i} U_t \psi_y \rangle - \langle \psi_x | \psi_y \rangle|$$

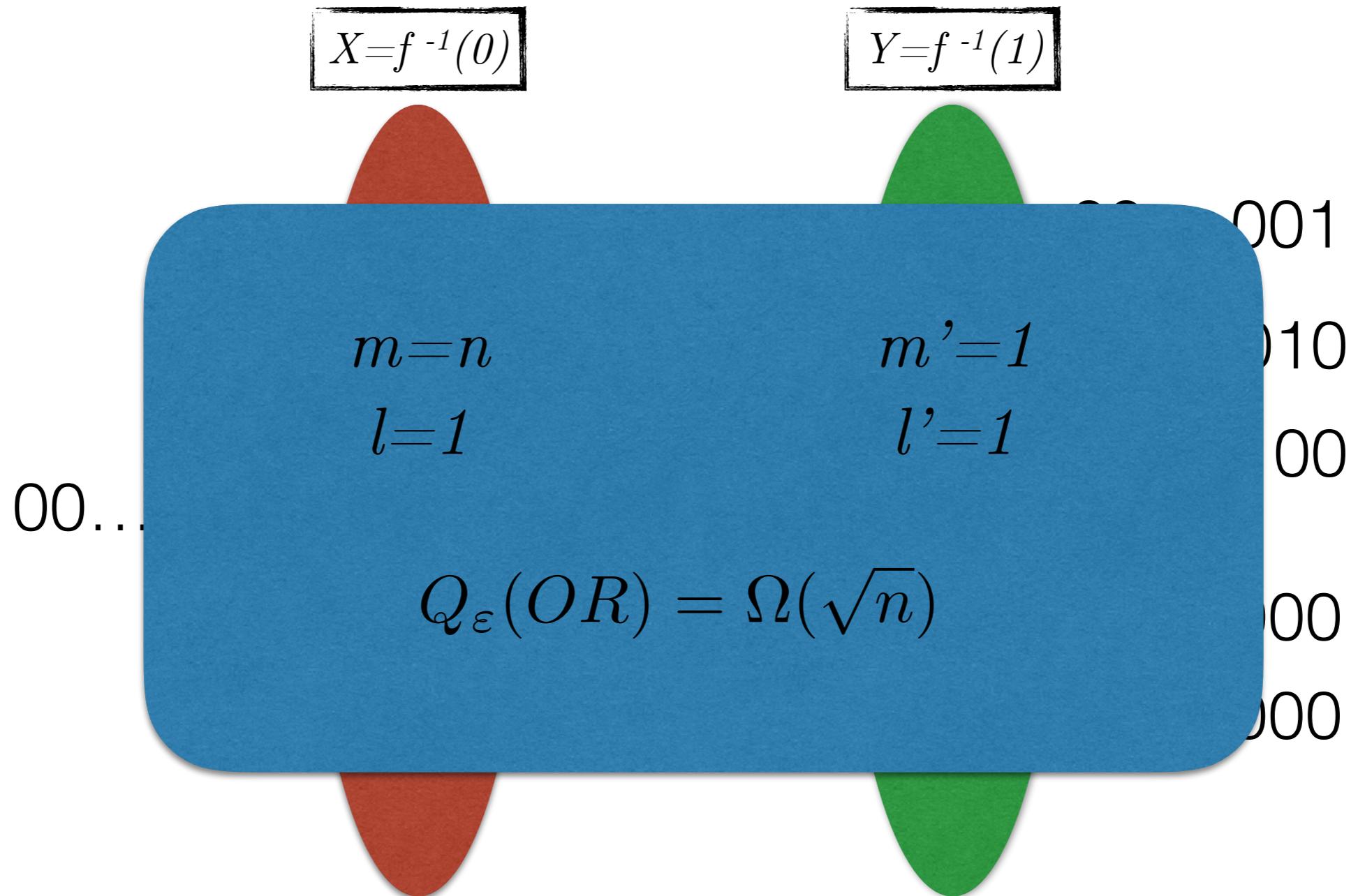
reduces to
counting edges
s.t. $x_i \neq y_i$

$$\leq \sqrt{|X||Y|ll'}$$

Application: the OR function



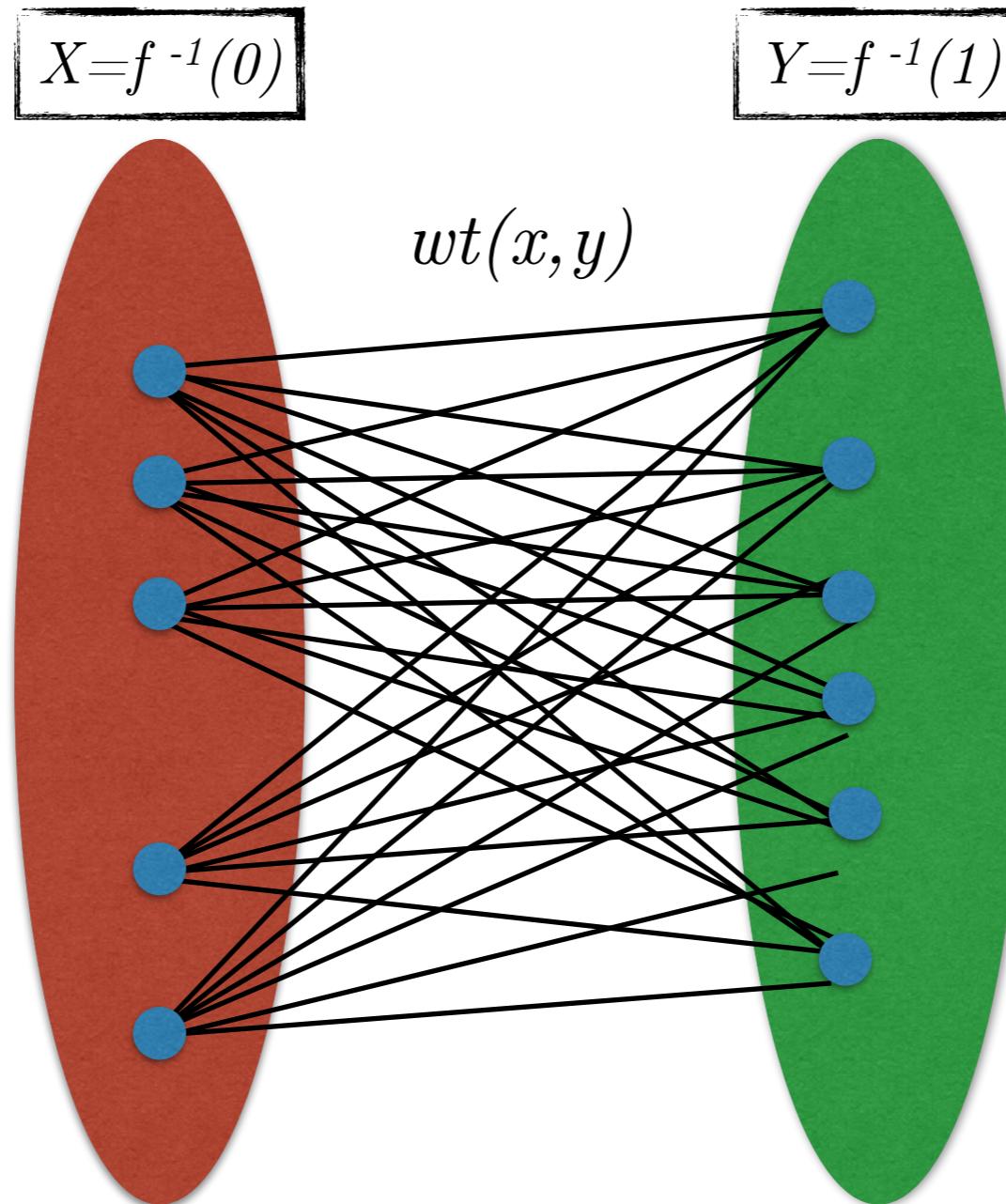
Application to the OR function



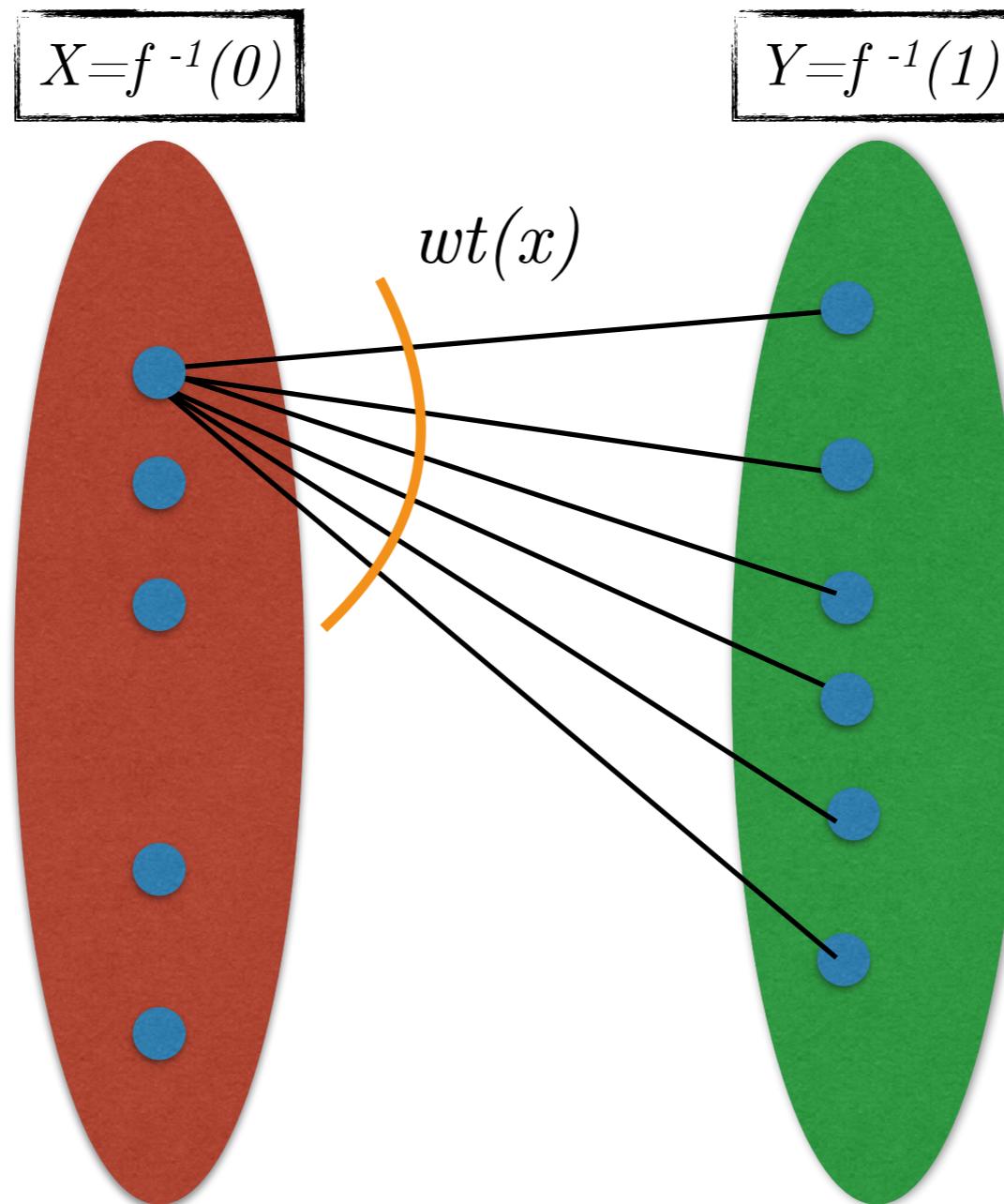
Adversary method(s)

- ✓ Original adversary method
- Weighted adversary method
- Spectral adversary method
- Generalized adversary method

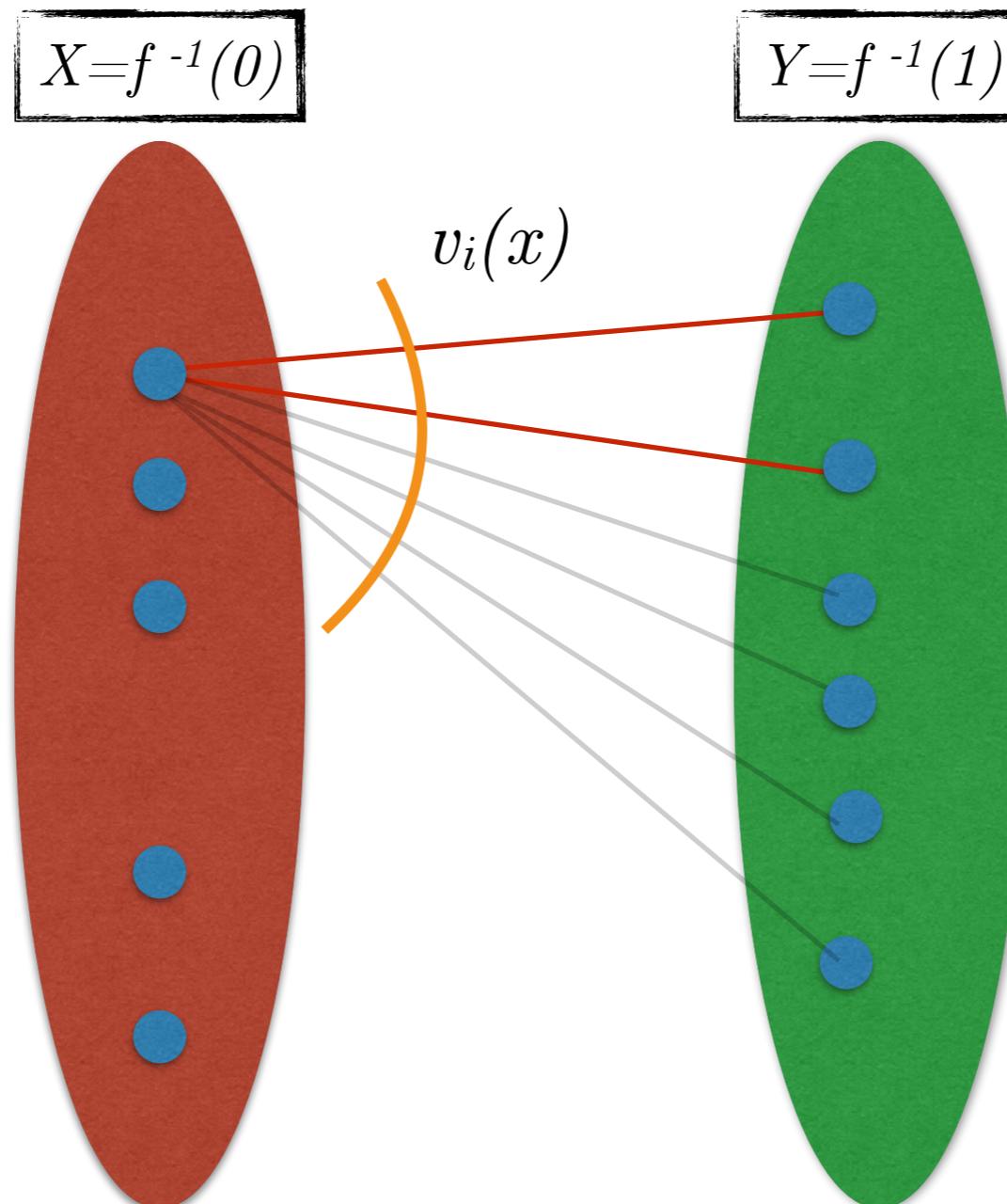
Weighted adversary method



Weighted adversary method



Weighted adversary method



$$x_i \neq y_i$$

Weighted adversary method

$$X = f^{-1}(0)$$

$$Y = f^{-1}(1)$$

$wt(x, y)$

$$Q_\varepsilon(f) = \max_{w, w'} \min_{\substack{x, y, i \\ w(x, y) > 0 \\ x_i \neq y_i}} \sqrt{\frac{wt(x)wt(y)}{v_i(x)v_i(y)}}$$

Adversary method(s)

- ✓ Original adversary method
- ✓ Weighted adversary method
- Spectral adversary method
- Generalized adversary method

Spectral adversary method

adversary matrix:

$$\Gamma \geq 0$$

$$\Gamma[x, y] = 0 \text{ iff } f(x) = f(y)$$

Masks :

$$D_i[x, y] = 1 \text{ iff } x_i \neq y_i$$

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

Spectral adversary method

adversary matrix:

$$\Gamma \geq 0$$

$$\Gamma[x, y] = 0 \text{ iff } f(x) = f(y)$$

Masks :

$$D_i[x, y] = 1 \text{ iff } x_i \neq y_i$$

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

Entry-wise
product

Spectral adversary method

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

$$Q_\varepsilon(f) = \Omega(Adv(f))$$

Spectral adversary method

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

- Easy generalization
- semi-definite program
- Ambainis' function f :

$$Q_\varepsilon(f) = \Omega(\deg(f)^{1.321})$$

Limitation of the adversary method

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

$$Adv(ED) \leq \sqrt{n}$$

$$Q_\varepsilon(ED) = \Omega(n^{2/3})$$

Certificate complexity barrier:
A feasible solution to the dual SDP

Adversary method(s)

- ✓ Original adversary method
- ✓ Weighted adversary method
- ✓ Spectral adversary method
- Generalized adversary method

Generalized adversary method

$$Adv(f) = \max_{\Gamma \geq 0} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

$$Adv^\pm(f) = \max_{\Gamma} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

Adv^\pm breaks the certificate complexity barrier
« Negative weight makes adversary stronger »
Høyer, Lee, Špalek

Generalized adversary method

$$Adv^\pm(f) = \max_{\Gamma} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

$$Q_\varepsilon(f) = \Theta(Adv^\pm(f))$$

Lower bound: Høyer, Lee, Špalek [2006]

Upper bound: Reichardt [09]

Dual of Adv^\pm can be interpreted as a span program

Generalized adversary method

$$Adv^\pm(f) = \max_{\Gamma} \frac{\|\Gamma\|}{\max_i \|\Gamma \circ D_i\|}$$

$$Adv^\pm(ED) = \Theta(n^{2/3})$$

Belovs [12]: explicit adversary matrix

Function composition

$$f: B^n \rightarrow C, g: A \rightarrow B$$

$$f \circ g^n: A^n \rightarrow C$$

$$f \circ g^n(x_1, \dots, x_n) = f(g(x_1), \dots, g(x_n))$$

$$Q_\varepsilon(f \circ g^n) = O(Q_\varepsilon(f)Q_\varepsilon(g))$$

Composition of algorithms: run an algorithm for f . For each query, run an algorithm for g .

Function composition

$$f: B^n \rightarrow C, g: A \rightarrow B$$

$$f \circ g^n: A^n \rightarrow C$$

$$f \circ g^n(x_1, \dots, x_n) = f(g(x_1), \dots, g(x_n))$$

If $B=\{0,1\}$:
$$Q_\varepsilon(f \circ g^n) = \Omega(Q_\varepsilon(f)Q_\varepsilon(g))$$

Easy counterexample to generalization:

g always returns an odd number

f is constant if all inputs are odd, hard otherwise

Proof overview

Appendix: Quantum Query Complexity

In our protocols, the work of the different parties is quantified by the number of queries made to black-box random functions, which can be modeled by a binary random oracle. In this Appendix, we review the main results from quantum query complexity that we used to prove our results and we sketch a new technical result that is useful for our lower bound proofs.

Upper Bounds

Our attacks can be modeled as quantum walks on Johnson graphs. The graph $J(n,r)$ is an undirected graph in which each node contains $\binom{n}{r}$ distinct elements of $[n]$ and every edge connects two nodes if and only if they differ by exactly two elements. Intuitively, we may think of “walking” from one node to an adjacent node by dropping one element and replacing it by another. The task is to find a specific element of $[n]$. The nodes that contain this element are called marked.

A random walk P on a Johnson graph can be quantized and the cost of the resulting quantum algorithm can be written as a function of S , U and C . These are the cost of setting up the quantum register in a state that corresponds to the stationary distribution, moving uniformly from one node to an adjacent node, and checking if a node is marked in order to flip its phase if it is, respectively.

Theorem 6. [28, 38] Let M be either empty, or the set of vertices that contain a fixed subset of elements $k \subseteq r$. Then there is a quantum algorithm that finds, with high probability, the k -subset of M if M is not empty and in the order of

$$S + \frac{1}{\sqrt{2}} \left(\frac{1}{\sqrt{r}} (|S| - k) + C \right).$$

where $S = n/r$ is the symmetric gap of the symmetric walk on $J(n,r)$ and $r = \binom{n}{r}$ is the probability that a random node is marked.

Lower Bounds

The central technical part of our lower bound consists in analyzing the complexity of a function closely related to the hardness of breaking the key establishment protocol. This function is obtained by composing element distinctions and a variant of the search problem. Recall that X denotes $X \cup [k]$, where X is an arbitrary set of integers.

Consider two integer parameters s and t and three functions $\alpha: [s] \rightarrow [t]$, $\nu: [s] \rightarrow [t]$ and $\lambda: [s] \times [t] \rightarrow [t]$ so that there exists a single pair (i,j) , $1 \leq i < j \leq s$, for which $\alpha(i) = \alpha(j)$, which is called a collision. The task is to find the unique nonzero collision in λ , having only access to a black-box that computes λ . This can be thought of as searching among t possibilities for the sole nonzero λ_{ij} .

10

10

Claim 2. $|\Gamma_\theta + \Gamma_\delta| = |\Gamma_\theta + \Gamma_\delta| \cdot |\Gamma_\theta + \Delta_\theta| \cdot \prod_{i \in I_\theta} |\Lambda_i|$.

Proof of Claim 2. Restricting to the block labelled by δ and $\tilde{\delta}$, Ref. [38] shows that

$$\text{ADV}^*(\theta, \Gamma_\theta) = |\Gamma_\theta + \Gamma_\delta| |\delta, \tilde{\delta}| \cdot |\Gamma_\theta + \Gamma_{\tilde{\delta}}|^{|\tilde{\delta}| - |\delta|} = \left(\bigotimes_{i \in I_\theta} \Gamma_i^{|\tilde{\delta}| - |\delta|} \right). \quad (2)$$

Here we use the second property of pSEARCH: for each q , there exist matrices Δ_q and Δ'_q such that when restricted to blocks $\Gamma_q = (\Gamma_\theta - \Gamma_\delta) \cap (\Delta_q + \Gamma_\delta \Delta'_q)$. Therefore, $\Gamma_q + \Gamma_\delta$ has the same block structure as Γ_θ and by Claim 2, we get the expression for $|\Gamma_\theta + \Gamma_\delta|$ given in Claim 2. \square

Equation 2 follows from Claims 1 and 2.

$$\begin{aligned} \text{ADV}^*(\theta, \Gamma_\theta) &= \min_{\delta \in [s]} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \\ &= \min_{\delta \in [s]} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \\ &\geq \min_{\delta \in [s]} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} \frac{|\Gamma_\theta|}{|\delta|} = |\Gamma_\theta| \cdot |\Gamma_\theta| = |\Gamma_\theta|^2. \end{aligned}$$

From the fact that $|\Gamma_\theta| = |M - \{k\}|$ and $|\Gamma_\theta + \Gamma_\delta| = |M - \{k\} \cup \Delta_\theta|$, it follows that

$$\text{ADV}^*(\theta, \Gamma_\theta) = \min_{\delta \in [s]} \frac{|\Gamma_\theta|}{|\delta|} \cdot \frac{|\Gamma_\theta|}{|\delta|}, \quad (3)$$

and therefore

$$\text{ADV}^*(\theta, \Gamma_\theta) \geq \text{ADV}^*(\theta) \cdot \min_{\delta \in [s]} \text{ADV}^*(\theta, \Gamma_\delta).$$

Proof of Claim 3. We prove $|\Gamma_\theta| \leq |\Gamma_\theta| \cdot \prod_{i \in I_\theta} |\Lambda_i|$. The proof proceeds in four steps.

1. We define a set of vectors $\{\Lambda_{i,j}\}$ in $\mathbb{C}^{|\Lambda_i||\Lambda_j|}$.
2. We prove that they are eigenvectors of Γ_θ and give the corresponding eigenvalues.
3. We show that we have defined all eigenvectors and eigenvalues of Γ_θ .
4. We upper bound the eigenvalues in absolute value.

Similarly to the way we built up Γ_θ from Γ_δ and the Γ_δ , we construct eigenvectors for Γ_θ using the eigenvectors for Γ_δ and the Λ_i as building blocks. We need some more notation before starting the proof. The spectrum of Λ_i is $[\Lambda_{i,1}, \Lambda_{i,2}]$ with eigenvalues $[\Lambda_{i,1}] \gg \dots \gg [\Lambda_{i,2}]$. For $\Lambda_i, \Lambda_j \in A$, we use the following notation:

$$\Lambda_{i,j}^{(k)} = \begin{cases} \Lambda_{i,j} & \text{if } i_k \neq j_k, \\ |\Lambda_i| & \text{otherwise.} \end{cases}$$

11

for each i and then finding two of those elements, among n possibilities, that are not distinct. Our main technical lemma, below, gives a lower bound on the number of queries to δ that are required. Lemma 5. Finding a nonzero collision in δ , structured as above, requires $\Omega(n^{2/3}q^{1/3})$ quantum queries to δ , except with vanishing probability.

It is more convenient to prove this lower bound for the related decision problem: we are given a function δ of the type above, but we are either given a function δ that has a single collision (or above) or on a non-collision function δ (in which case it is collision-free, except for value 0 in its image). The task is to decide which is the case. Obviously, any algorithm that can solve the search problem with probability of success at least $p > 0$ can be used to solve the decision problem with error bounded by $\frac{1}{2} - p$: run the search algorithm; if a collision is found (and certified), output “collision”, otherwise output either “collision” or “no collision” with equal probability after flipping a fair coin. This follows that any lower bound on the bounded-error decision problem applies equally well to the search problem.

We shall change the notation in order to adapt it to the normal usage in the field of quantum query complexity. The function $v: [n] \rightarrow [n]$ is represented by an element of $[n]^n$. This makes it possible to think of the decision version of element distinction as a Boolean function $\text{ED}: [n]^n \rightarrow \{0, 1\}$, although it is a partial function since there is a promise on the valid inputs to ED : given n integers $\{i_1, \dots, i_n\} \in [n]^n$, the promise is that all the elements are distinct or that all the elements are distinct except two, say $i_1 \neq i_2$. The goal is to decide which of the two cases occurs by making no fewer than ϵ queries to δ , for some constant $\epsilon > 0$.

Anchored element distinction quantum algorithm [2] runs in $O(n^{2/3})$ queries to the input, and Arora and Barak prove that this is optimal [2]. Although this lower bound was proven using the polynomial method [2], a similar proof of the [28] claim that the general adversary method is tight for total and partial functions follows our proof of the lower bound in decision using the generalized adversary method [34], we may conclude that there exists an $\Omega(n^{2/3})$ adversary bound for element distinction.

We compare the element distinction problem with a instance of a promise version of a search problem, which we call pSEARCH.

Definition 1. pSEARCH: $P \rightarrow A$ with $P \subseteq A^P$ is a promise problem. On input (i_1, \dots, i_n) , the promise P is that all but one of the values are zero. The goal is to find and output the nonzero value by making queries that take i as input and return a_i .

The composed function, with $A = \{A_i\}$, is denoted Γ_θ . On input $x \in P^*$,

$$\Gamma_\theta(x) = \text{ED}(\text{pSEARCH}(x_1), \dots, \text{pSEARCH}(x_n)).$$

We now prove that the quantum query complexity of H is in $\Omega(n^{2/3}q^{1/3})$. The proof uses the generalized adversary method for quantum query complexity, which we briefly describe here. Suppose we want to determine the quantum query complexity of a function F . First, we assign weights to pairs of inputs in order to bring out how hard it is (in terms of number of queries) to distinguish these inputs apart from one another. The adversary lower bound is the worst ratio of the spectral norm of this matrix, which measures the overall progress necessary in order for the algorithm to succeed, to the spectral norm of associated matrix G , which measures the maximum amount of progress that can be achieved by making a single query.

12

Definition 2. Fix a function $F: S \rightarrow T$. A symmetric matrix $\Gamma: S \times S \rightarrow \mathbb{R}$ is an adversary matrix for F provided $\Gamma(x, y) = 0$ whenever $F(x) = F(y)$. Let $\delta(x, y) = 1$ if $x_i \neq y_i$ and 0 otherwise. The adversary bound of F using Γ is

$$\text{ADV}^*(F, \Gamma) = \min_{\delta} \frac{|\Gamma|}{|\delta|}.$$

where δ denotes entropic (or Hadamard) product, and $|\Lambda|$ denotes the spectral norm of Λ (which is equal to its largest eigenvalue). The adversary bound $\text{ADV}^*(F)$ is the minimum, over all adversary matrices Γ for F , of $\text{ADV}^*(F, \Gamma)$.

Since H is defined as the composition of ED and pSEARCH, we would like to apply a composition theorem for the generalized adversary method [34], which would say that if a function $H = F \circ G$, then $\text{ADV}^*(H) \geq \text{ADV}^*(F) \cdot \text{ADV}^*(G)$. Unfortunately, the composition theorem of Ref. [34, 33] requires the inner (and outer) $\text{ADV}^*(\cdot)$ functions to be Boolean, which is not the case here for the inner function pSEARCH. Since counter-examples can be found, we cannot hope to prove a fully general composition theorem in which the inner function would be an arbitrary function. Nevertheless, we prove here a composition theorem with pSEARCH as the inner function.

Theorem 6. Let $F: A^P \rightarrow B$, $\text{pSEARCH}: P \rightarrow A$ with $P \subseteq A^P$ as described above, and $H = F \circ \text{pSEARCH}$. Then

$$\text{ADV}^*(H) \geq \frac{1}{2} \text{ADV}^*(F) \cdot \text{ADV}^*(\text{pSEARCH}).$$

The inner function can be slightly more general than pSEARCH. For example, it could be that the element function is given in blocks in several places. The proof also goes through if the instances of pSEARCH operate over distinct domains $|A_i^P|$. We leave for further research the extent to which our theorem can be generalized and proved to prove it as stated.

Proof. We prove the theorem using only a few properties of pSEARCH, which we describe below. In order to distinguish the i instances of pSEARCH, and to simplify notation, we write the inner function as $G_1, \dots, G_n: P \rightarrow A$ with $P \subseteq A^P$, $|A| = M$, and $|P| = M^P$. We use the fact that G_i is π -symmetric for $i \in [n]$. We assume that inputs are sorted according to the output value. We use the crucial properties of pSEARCH.

1. The $M \times M$ optimal adversary matrices Γ_θ for G_i can be written in block form with $M \times M$ blocks of size $q \times q$ indexed by pairs of outputs in which all off-diagonal blocks are identical. Written in this form, all $M \times M$ diagonal blocks are necessarily zero since it is an π -symmetric matrix.
2. The $M \times M$ matrix Γ_θ with inputs sorted in the same way, is also composed of identical off-diagonal blocks Δ_q and Δ_q^T on diagonal blocks. Notice that this strongly depends on G_i , since the inputs are sorted by output value.

For any function F , consider $H = F \circ G_1, \dots, G_n$. We show that for all adversary matrices Γ_θ for G_i the form $\Gamma_\theta = (\Gamma_{i,1} \otimes \dots \otimes \Gamma_{i,n})$. We know that the all adversary matrices Γ_θ for G_i are π -symmetric.

$$\text{ADV}^*(H) \geq \text{ADV}^*(F) \cdot \min_{\delta} \text{ADV}^*(G_i, \Gamma_\theta). \quad (1)$$

13

As we can see from the following eigenvalue equation, $\Lambda_{i,j}^{(k)}$ is the eigenvalue of $\Gamma_i^{(k)}$ associated with the vector $\Lambda_{i,j}$:

$$\begin{aligned} \Gamma_i^{(k)} \Lambda_{i,j} &= \begin{cases} \Lambda_{i,j} \Lambda_{i,j} & \text{if } i_k \neq j_k, \\ |\Lambda_i| \Lambda_{i,j} & \text{otherwise} \end{cases} \\ &= \Lambda_{i,j}^{(k)} \Lambda_{i,j}. \end{aligned}$$

Given a vector of indices $v = (v_1, \dots, v_n) \in [n]^n$, we build up our eigenvectors for Γ_θ by picking the $v_i^{(k)}$ eigenvalues for the i^{th} inner function (see Step 1). For $v = (v_1, \dots, v_n)$, the $M \times M$ matrix A_v is defined by blocks

$$A_v(x, y) = \Gamma_\theta(x, y) \prod_{i=1}^n v_i^{(x_i)},$$

and we write its spectrum:

$$\{|\Lambda_{i,j}| \}_{i,j \in I_\theta}.$$

Step 1. We are ready to define the eigenvectors $\Lambda_{i,j}$ of Γ_θ . We define the vectors $\Lambda_{i,j}$ on the block $\Gamma_i^{(k)}$ of coordinate $x \in P^*$ such that $(x_1, \dots, x_n, x_i) = v$:

$$\Lambda_{i,j}^{(k)} = \langle \delta | \delta \rangle \left(\bigotimes_{i=1}^n \Lambda_{i,j} \right).$$

Note that because of the structure of the $\Gamma_i^{(k)}$, it suffices for our purposes to build up the eigenvectors of Γ_θ from the eigenvectors of the underlying G_i , which considerably simplifies the proof.

Step 2. We claim that the $\Lambda_{i,j}$ are eigenvectors of Γ_θ with corresponding eigenvalues $\mu_{i,j}$. We want to calculate $\Gamma_\theta(\delta, \delta)$. We do this by block by block. Fix $x \in A^P$. Using the eigenvalue equation (2), we get

$$\bigotimes_i \Gamma_i^{(k)} \bigotimes_{i=1}^n \Lambda_{i,j} = \prod_{i=1}^n \Lambda_{i,j}^{(k)} \bigotimes_{i=1}^n \Lambda_{i,j}. \quad (2)$$

Then

$$\begin{aligned} (\Gamma_\theta \Lambda_{i,j})^{(k)} &= \sum_i \left(\Gamma_i(x, \delta) \bigotimes_{i=1}^n \Gamma_i^{(k)} \right) \left(\delta \bigotimes_{i=1}^n \Lambda_{i,j} \right) \\ &= \sum_i \Gamma_i(x, \delta) \prod_{i=1}^n \Lambda_{i,j}^{(k)} \bigotimes_{i=1}^n \Lambda_{i,j}, \text{ by Equation 2} \\ &= \sum_i \Lambda_{i,j}^{(k)} \delta \bigotimes_{i=1}^n \Lambda_{i,j} \\ &= \mu_{i,j} \delta \bigotimes_{i=1}^n \Lambda_{i,j} \\ &= \mu_{i,j} \Lambda_{i,j}. \end{aligned}$$

14

Step 3. We prove that the vectors $\Lambda_{i,j}$ span $\mathbb{C}^{|\Lambda_i||\Lambda_j|}$. There are q^n matrices A_v , and only one has q^n eigenvectors α . Therefore, $\{\Lambda_{i,j}\}$ is a collection of $(qM)^n$ vectors. We now prove that they are orthogonal. Notice that

$$\begin{aligned} \langle \Lambda_{i,j}, \Lambda_{i,j'} \rangle &= \sum_{x \in P^*} \langle \Lambda_{i,j}^{(k)}, \Lambda_{i,j'}^{(k)} \rangle \\ &= \sum_{x \in P^*} \left(\langle \delta | \delta \rangle \langle \delta | \delta \rangle \prod_{i=1}^n \langle \Lambda_{i,j}, \Lambda_{i,j'} \rangle \right) \\ &= \langle \delta | \delta \rangle \prod_{i=1}^n \langle \Lambda_{i,j}, \Lambda_{i,j'} \rangle. \end{aligned}$$

If $\Lambda_{i,j} \neq \Lambda_{i,j'}$, it must be the case that either $v \neq v'$ or $\alpha \neq \alpha'$. Assume $v \neq v'$. Then for some i , $v_{i,j} \neq v_{i,j'}$ and since these vectors form an orthonormal basis of \mathbb{C}^n , we get $\langle \Lambda_{i,j}, \Lambda_{i,j'} \rangle = 0$. Now if $\alpha \neq \alpha'$, then $v = v'$. Again, these vectors form an orthonormal basis of \mathbb{C}^{qM} and we get $\langle \alpha, \alpha' \rangle = 0$.

Step 4. We prove by induction that the eigenvalues $\mu_{i,j}$ of Γ_θ are such that $|\mu_{i,j}| \leq |\Lambda_{i,j}| \leq |\Gamma_\theta| \prod_{i=1}^n |\Lambda_i|$ for all i and j . For $i \in [n]$ and $\alpha \in \mathbb{C}^q$, we define a family of matrices $A_\alpha^{(k)}$ inductively as follows:

1. $A_\alpha^{(k)} = \Gamma_\theta$.
2. $A_\alpha^{(k)}[x, y] = A_\alpha^{(k-1)}[x, y] \cdot \Lambda_{i,j}^{(k)}$.

By definition, $A_\alpha^{(0)} = \Lambda_i$. We prove by induction that for each i ,

$$|\Lambda_{i,j}| \leq |\Gamma_\theta| \leq |\Gamma_\theta| \prod_{i=1}^n |\Lambda_i|.$$

Since $\mu_{i,j}$ is an eigenvalue of A_α , this implies $|\mu_{i,j}| \leq |\Lambda_{i,j}| \leq |\Gamma_\theta| \prod_{i=1}^n |\Lambda_i|$.

Since $A_\alpha^{(k)}$ is Γ_θ , the base

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad} C$$
$$x = (x_1, \dots, x_n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\Gamma_g = \left(\begin{array}{c|c} 0 & \Gamma_g^{1,0} \\ \hline \Gamma_g^{0,1} & 0 \end{array} \right) \qquad \Gamma_f = \left(\begin{array}{c} \Gamma_f[\tilde{x}, \tilde{y}] \end{array} \right)$$

$$\Gamma_{f \circ g^n} = \left(\begin{array}{c|c|c} & & \\ \hline & \Gamma_{\tilde{x}, \tilde{y}} & \\ \hline & & \end{array} \right)$$

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad f \quad} C$$
$$x = (x_1, \dots, x_n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\tilde{x} = (g(x_1), g(x_2), \dots, g(x_n))$$

$$\Gamma^{\tilde{x}, \tilde{y}} = \Gamma_f[\tilde{x}, \tilde{y}].$$

$$\Gamma_g = \left(\begin{array}{c|c} 0 & \Gamma_g^{1,0} \\ \hline \Gamma_g^{0,1} & 0 \end{array} \right)$$

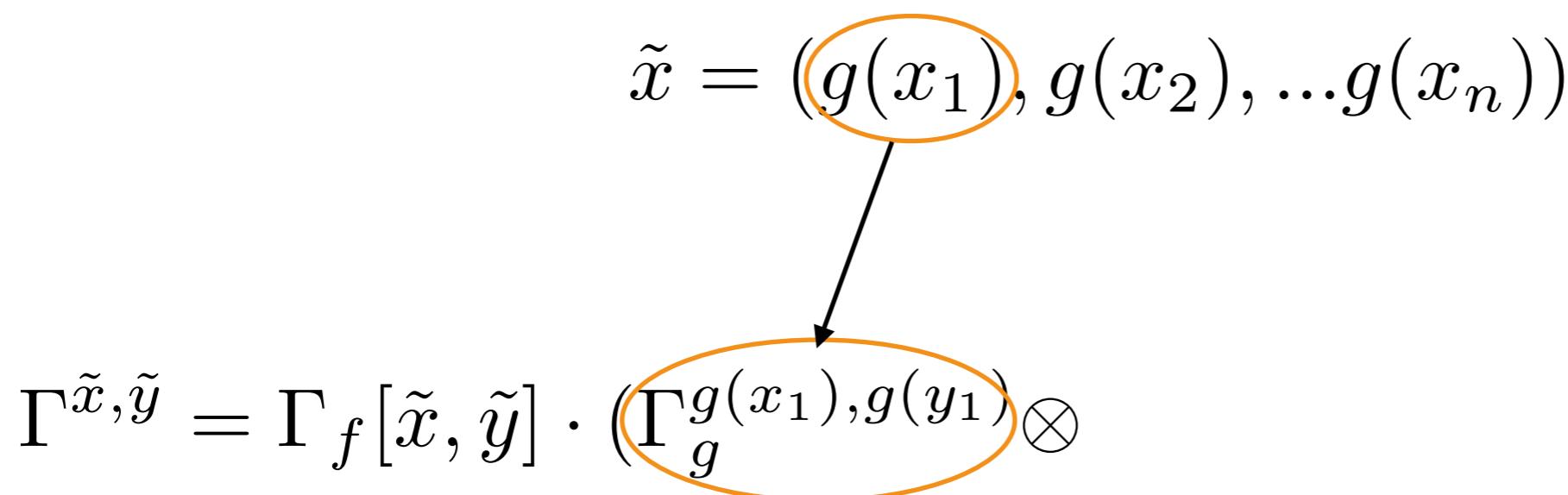
$$\Gamma_f = \left(\begin{array}{c} \Gamma_f[\tilde{x}, \tilde{y}] \end{array} \right)$$

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad f \quad} C$$

$$x = (x_1, \dots, x_n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\tilde{x} = (g(x_1), g(x_2), \dots, g(x_n))$$



$$\Gamma_{\tilde{x}, \tilde{y}} = \Gamma_f[\tilde{x}, \tilde{y}] \cdot (\Gamma_g^{g(x_1), g(y_1)} \otimes \dots)$$

$$\Gamma_g = \left(\begin{array}{c|c} 0 & \Gamma_g^{1,0} \\ \hline \Gamma_g^{0,1} & 0 \end{array} \right)$$

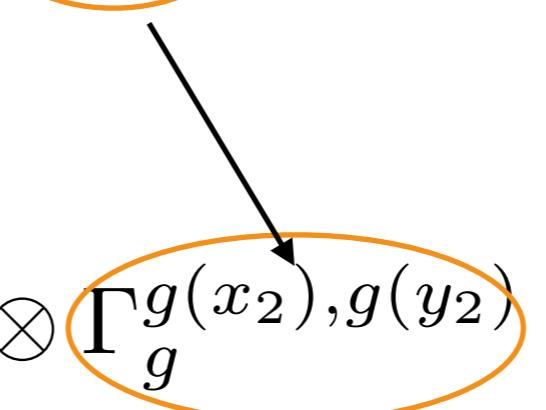
$$\Gamma_f = \left(\begin{array}{c} \Gamma_f[\tilde{x}, \tilde{y}] \end{array} \right)$$

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad f \quad} C$$

$$x = (x_1, \dots, x_n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\tilde{x} = (g(x_1), g(x_2), \dots, g(x_n))$$

$$\Gamma^{\tilde{x}, \tilde{y}} = \Gamma_f[\tilde{x}, \tilde{y}] \cdot (\Gamma_g^{g(x_1), g(y_1)} \otimes \Gamma_g^{g(x_2), g(y_2)})$$


$$\Gamma_g = \left(\begin{array}{c|c} 0 & \Gamma_g^{1,0} \\ \hline \Gamma_g^{0,1} & 0 \end{array} \right)$$

$$\Gamma_f = \left(\begin{array}{c} \Gamma_f[\tilde{x}, \tilde{y}] \end{array} \right)$$

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad f \quad} C$$

$$x = (x_1, \dots, x_n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\tilde{x} = (g(x_1), g(x_2), \dots, g(x_n))$$

$$\Gamma^{\tilde{x}, \tilde{y}} = \Gamma_f[\tilde{x}, \tilde{y}] \cdot (\Gamma_g^{g(x_1), g(y_1)} \otimes \Gamma_g^{g(x_2), g(y_2)} \otimes \dots \otimes \Gamma_g^{g(x_n), g(y_n)})$$

$$\Gamma_g = \left(\begin{array}{c|c} 0 & \Gamma_g^{1,0} \\ \hline \Gamma_g^{0,1} & 0 \end{array} \right)$$

$$\Gamma_f = \left(\begin{array}{c} \Gamma_f[\tilde{x}, \tilde{y}] \end{array} \right)$$

Function composition

$$A^m \xrightarrow{\quad} \{0,1\}^n \xrightarrow{\quad} C$$
$$x = (x^1, \dots, x^n) \xrightarrow{g} \tilde{x} = g^n(x) \xrightarrow{f} f(\tilde{x}) = f \circ g^n(x)$$

$$\Gamma_{f \circ g^n} = \left(\begin{array}{c|c|c} \hline & & \\ \hline & \Gamma^{\tilde{x}, \tilde{y}} & \\ \hline & & \end{array} \right) \qquad \Gamma^{\tilde{x}, \tilde{y}} = \Gamma_f[\tilde{x}, \tilde{y}] \cdot \left(\bigotimes_{i=1}^m \overline{\Gamma}_g^{\tilde{x}_i, \tilde{y}_i} \right)$$

$$\|\Gamma_{f \circ g^m}\| = \|\Gamma_f\| \|\Gamma_g\|^m$$

And something similar for $\Gamma \circ D_i \dots$

Multiplicative adversary

- Better handling of the errors
- Used to prove direct product theorems (rather than direct sum)
- Also generalizes the polynomial method

Other applications

- Various functions: graph problems, Element Distinctness, etc...
- Cryptographic applications: Maximum gap is polynomial !
 - Merkle Puzzles
 - Symmetric cryptography