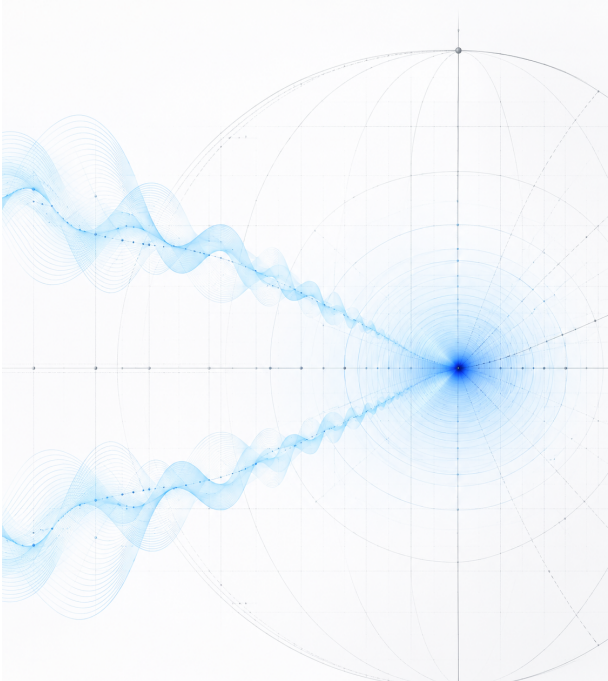
LEZIONE 19

L’algoritmo di Deutsch–Jozsa

Da un ingresso binario a *n* qubit



Metodi della Fisica Quantistica per l’Ingegneria

Prof. Vincenzo Berardi · Politecnico di Bari

Da Deutsch a Deutsch–Jozsa

Nella Lect18 abbiamo distinto le due classi di funzioni $f : \{0,1\} \to \{0,1\}$ con una sola interrogazione quantistica. Ora il registro d'ingresso contiene n qubit:

$$\boxed{f : \{0,1\}^n \longrightarrow \{0,1\}.}$$

Il dominio ha 2^n elementi. La domanda non cambia:

la funzione è costante oppure bilanciata?

L'algoritmo di Deutsch–Jozsa estende a n qubit il meccanismo di **codifica nelle fasi** e **interferenza**.

La promessa sul tipo di funzione

Il problema garantisce che valga *una* delle condizioni:

1. **Costante:** $f(x) = 0$ per tutti gli input, oppure $f(x) = 1$ per tutti gli input.
2. **Bilanciata:** esattamente 2^{n-1} ingressi danno 0 e altrettanti danno 1.

Una funzione generica potrebbe non appartenere a nessuna delle due classi: la *promise* è quindi un'ipotesi indispensabile.

Il compito è identificare la classe, non ricostruire la tabella completa di f.

Esempio: otto ingressi

Per $n = 3$ sono possibili $2^3 = 8$ stringhe d'ingresso:

$$000,\ 001,\ 010,\ 011,\ 100,\ 101,\ 110,\ 111.$$

x	000	001	010	011	100	101	110	111
$f_{\mathrm{c}}(x)$	0	0	0	0	0	0	0	0
$f_{\mathrm{b}}(x)$	0	0	0	0	1	1	1	1

La prima funzione è costante; la seconda è bilanciata. L'ordine degli zeri e degli uni non è vincolato dalla promessa.

Quante query servono classicamente?

Consideriamo algoritmi **classici deterministici ed esatti**. Dopo 2^{n-1} interrogazioni con lo stesso risultato, sono ancora possibili:

$$\underbrace{0,\ldots,0}_{2^{n-1}} \quad \begin{cases} \underbrace{0,\ldots,0}_{2^{n-1}} & \text{costante,} \\ \underbrace{1,\ldots,1}_{2^{n-1}} & \text{bilanciata.} \end{cases}$$

Una query ulteriore è necessaria nel caso peggiore ed è sufficiente:

$$\boxed{Q_{\mathrm{C}}^{\mathrm{det,exact}}(n) = 2^{n-1} + 1.}$$

Per $n = 3$: cinque query; per $n = 10$: 513 query.

La risorsa quantistica: due registri

Si impiegano n qubit di ingresso e un qubit ausiliario (*ancilla*).

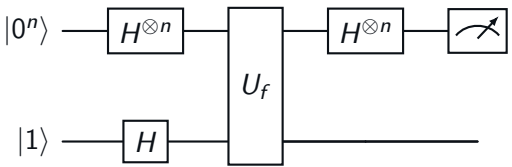
Registro	Numero di qubit	Stato iniziale
Ingresso	n	$\|0\rangle^{\otimes n} = \|0^n\rangle$
Ancilla	1	$\|1\rangle$

Lo stato complessivo è il prodotto tensoriale:

$$\boxed{|\psi_0\rangle = |0^n\rangle\,|1\rangle\,.}$$

Il risultato finale sarà ricavato dalla misura del **solo registro di ingresso**.

Il circuito di Deutsch–Jozsa



La linea superiore rappresenta **un registro di n qubit**; $H^{\otimes n}$ indica una Hadamard su ciascun qubit.

1. Preparazione della sovrapposizione uniforme.
2. Una sola applicazione dell'oracle U_f.
3. Hadamard finali e misura dei n qubit di ingresso.

Preparazione: una Hadamard per ogni qubit

Per un singolo qubit:

$$H|0\rangle = |+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}.$$

Per n qubit inizialmente in $|0^n\rangle$:

$$H^{\otimes n}|0^n\rangle = \underbrace{|+\rangle \otimes \cdots \otimes |+\rangle}_{n \text{ fattori}}$$

$$= \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle .$$

Ogni stringa x compare con ampiezza $1/\sqrt{2^n}$.

Preparazione: esempio con $n = 2$

Si ha $|0^2\rangle = |00\rangle$; dunque

$$(H \otimes H)\,|00\rangle = H\,|0\rangle \otimes H\,|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

Le quattro ampiezze sono identiche e la probabilità di ogni ingresso è $1/4$.

Preparazione dell'ancilla

L'ancilla parte da $|1\rangle$ e riceve una Hadamard:

$$H|1\rangle = |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Combinando i due registri, dopo il primo livello di porte:

$$\boxed{|\psi_1\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_x |x\rangle\right) \otimes |-\rangle = \frac{1}{\sqrt{2^n}} \sum_x |x\rangle\,|-\rangle\,.}$$

Il primo registro è in sovrapposizione uniforme; l'ancilla è nello stato $|-\rangle$.

L'oracle resta quello già definito

Per $x \in \{0,1\}^n$ e $y \in \{0,1\}$:

$$\boxed{U_f \left|x\right\rangle \left|y\right\rangle = \left|x\right\rangle \left|y \oplus f(x)\right\rangle .}$$

Se $f(x) = 0$, il secondo registro non cambia; se $f(x) = 1$, $|0\rangle$ e $|1\rangle$ si scambiano. L'azione dell'oracle su $|-\rangle$ permette di trasformare il valore di $f(x)$ in un segno.

Phase kickback: i due casi

Applichiamo la definizione di U_f allo stato dell'ancilla:

$$\begin{aligned} U_f |x\rangle |-\rangle &= \frac{U_f |x\rangle |0\rangle - U_f |x\rangle |1\rangle}{\sqrt{2}} \\ &= \frac{|x\rangle |f(x)\rangle - |x\rangle |1 \oplus f(x)\rangle}{\sqrt{2}}. \end{aligned}$$

Se $f(x) = 0$**:** lo stato è $|x\rangle (|0\rangle - |1\rangle)/\sqrt{2} = |x\rangle |-\rangle$. **Se** $f(x) = 1$**:** lo stato è $|x\rangle (|1\rangle - |0\rangle)/\sqrt{2} = -|x\rangle |-\rangle$.

Phase kickback: formula generale

I due casi precedenti si riassumono nella relazione:

$$\boxed{U_f \left|x\right\rangle \left|-\right\rangle = (-1)^{f(x)} \left|x\right\rangle \left|-\right\rangle .}$$

Il primo registro non cambia stringa; il valore di $f(x)$ determina il **segno dell'ampiezza**.

$$(-1)^{f(x)} = \begin{cases} +1, & f(x) = 0, \\ -1, & f(x) = 1. \end{cases}$$

Non viene effettuata alcuna misura durante l'applicazione dell'oracle.

Una sola query sull'intera sovrapposizione

Partiamo da $|\psi_1\rangle = 2^{-n/2} \sum_x |x\rangle\,|-\rangle$. Per linearità dell'operatore U_f:

$$|\psi_2\rangle = U_f \left(\frac{1}{\sqrt{2^n}} \sum_x |x\rangle\,|-\rangle \right) = \frac{1}{\sqrt{2^n}} \sum_x U_f\,|x\rangle\,|-\rangle$$

$$= \boxed{\frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)}\,|x\rangle\,|-\rangle\,.}$$

L'oracle è stato applicato **una sola volta**, anche se la sovrapposizione contiene 2^n componenti.

Che cosa sappiamo dopo l'oracle?

Il registro di ingresso è nello stato

$$\boxed{|\phi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} |x\rangle \,.}$$

Tutte le ampiezze hanno ancora modulo $1/\sqrt{2^n}$. La misura immediata di x darebbe quindi

$$P(x) = \left| \frac{(-1)^{f(x)}}{\sqrt{2^n}} \right|^2 = \frac{1}{2^n},$$

indipendentemente da $f(x)$. L'informazione utile è nelle **fasi relative**: dobbiamo trasformarla in probabilità misurabili.

Una Hadamard: origine dei segni

Partiamo dall'azione nota della porta Hadamard:

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Per $x, y \in \{0, 1\}$, esaminiamo il coefficiente di $|y\rangle$ nello stato $H|x\rangle$:

x	y	coefficiente di $\lvert y\rangle$	$(-1)^{xy}$
0	0	$+1/\sqrt{2}$	$+1$
0	1	$+1/\sqrt{2}$	$+1$
1	0	$+1/\sqrt{2}$	$+1$
1	1	$-1/\sqrt{2}$	-1

Il segno meno compare soltanto per $x = y = 1$: è l'elemento $H_{11} = -1/\sqrt{2}$.

Una Hadamard: formula unica per $|0\rangle$ e $|1\rangle$

La tabella precedente mostra che il fattore $(-1)^{xy}$ produce il segno corretto per ogni coppia di bit (x, y). Quindi possiamo scrivere l'azione della porta come somma sui due possibili stati di uscita:

$$\boxed{H|x\rangle = \frac{1}{\sqrt{2}} \sum_{y=0}^{1} (-1)^{xy} |y\rangle .}$$

Controlliamo entrambi gli ingressi:

$$x = 0: \quad H|0\rangle = \frac{(-1)^0|0\rangle + (-1)^0|1\rangle}{\sqrt{2}} = \frac{|0\rangle + |1\rangle}{\sqrt{2}},$$

$$x = 1: \quad H|1\rangle = \frac{(-1)^0|0\rangle + (-1)^1|1\rangle}{\sqrt{2}} = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

La formula riassume le due relazioni note; non introduce una nuova porta.

Due qubit: applichiamo una Hadamard a ciascuno

Consideriamo lo stato $|x_1 x_2\rangle = |x_1\rangle \otimes |x_2\rangle$. Per definizione del prodotto tensoriale di operatori,

$$(H \otimes H)\,|x_1 x_2\rangle = H\,|x_1\rangle \otimes H\,|x_2\rangle\,.$$

Sostituiamo in ciascun fattore la formula appena ricavata:

$$\begin{aligned}(H \otimes H)\,|x_1 x_2\rangle &= \left[\frac{1}{\sqrt{2}} \sum_{y_1=0}^{1} (-1)^{x_1 y_1}\,|y_1\rangle\right] \otimes \left[\frac{1}{\sqrt{2}} \sum_{y_2=0}^{1} (-1)^{x_2 y_2}\,|y_2\rangle\right] \\ &= \frac{1}{2}\left[\sum_{y_1=0}^{1} (-1)^{x_1 y_1}\,|y_1\rangle\right] \otimes \left[\sum_{y_2=0}^{1} (-1)^{x_2 y_2}\,|y_2\rangle\right].\end{aligned}$$

I due fattori $1/\sqrt{2}$ si moltiplicano e producono $1/2$.

Due qubit: distribuiamo il prodotto tensoriale

Il prodotto fra le due somme associa ogni $y_1 \in \{0,1\}$ a ciascun $y_2 \in \{0,1\}$; inoltre $|y_1\rangle \otimes |y_2\rangle = |y_1 y_2\rangle$.

$$(H \otimes H)\,|x_1 x_2\rangle = \frac{1}{2} \sum_{y_1=0}^{1} \sum_{y_2=0}^{1} (-1)^{x_1 y_1} (-1)^{x_2 y_2}\,|y_1 y_2\rangle$$

$$= \boxed{\frac{1}{2} \sum_{y_1, y_2=0}^{1} (-1)^{x_1 y_1 + x_2 y_2}\,|y_1 y_2\rangle\,.}$$

La somma contiene quattro termini, uno per ogni possibile uscita $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$. I segni si moltiplicano: $(-1)^a (-1)^b = (-1)^{a+b}$.

Da due a n qubit: il prodotto delle somme

La stessa costruzione vale per la stringa $x = x_1 \cdots x_n$:

$$H^{\otimes n} |x\rangle = \bigotimes_{j=1}^{n} H |x_j\rangle$$

$$= \bigotimes_{j=1}^{n} \left[\frac{1}{\sqrt{2}} \sum_{y_j=0}^{1} (-1)^{x_j y_j} |y_j\rangle \right] .$$

Sviluppando i prodotti tensoriali, scegliamo un valore y_j per ogni qubit:

$$\boxed{H^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y_1,\ldots,y_n=0}^{1} \left[\prod_{j=1}^{n} (-1)^{x_j y_j} \right] |y_1 \cdots y_n\rangle .}$$

Il fattore $1/\sqrt{2^n}$ proviene dai n coefficienti $1/\sqrt{2}$; la somma contiene 2^n termini.

Da due a n qubit: come si combinano i segni

Il coefficiente di ogni stringa di uscita è il prodotto di n segni:

$$\prod_{j=1}^{n}(-1)^{x_j y_j} = (-1)^{x_1 y_1 + \cdots + x_n y_n} = (-1)^{\sum_{j=1}^{n} x_j y_j}.$$

Scrivendo $y = y_1 \cdots y_n$ e sommando su tutte le stringhe binarie,

$$\boxed{H^{\otimes n}\,|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} (-1)^{\sum_{j=1}^{n} x_j y_j}\,|y\rangle\,.}$$

Questa è l'azione di $H^{\otimes n}$ sulla base computazionale, ricavata direttamente dalla formula di Hadamard per un qubit.

Prodotto scalare binario: abbreviazione della formula

Nella formula precedente compare la somma di interi

$$S(x,y) = \sum_{j=1}^{n} x_j y_j.$$

Il segno $(-1)^S$ dipende dalla parità di S (pari: $+1$; dispari: -1). Definiamo il **prodotto scalare binario**:

$$x \cdot y = \left(\sum_{j=1}^{n} x_j y_j \right) \bmod 2, \qquad (-1)^{S(x,y)} = (-1)^{x \cdot y}.$$

Di conseguenza, la formula si abbrevia in

$$\boxed{H^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle .}$$

Esempio: prodotto scalare binario

Per $x = 101$ e $y = 110$:

$$x \cdot y = (1 \cdot 1 + 0 \cdot 1 + 1 \cdot 0) \bmod 2 = 1,$$
$$(-1)^{x \cdot y} = -1.$$

Per $x = 101$ e $y = 101$:

$$x \cdot y = (1 \cdot 1 + 0 \cdot 0 + 1 \cdot 1) \bmod 2 = 0,$$
$$(-1)^{x \cdot y} = +1.$$

Il prodotto scalare binario non è il consueto prodotto scalare reale: il risultato viene ridotto modulo 2.

Le Hadamard finali: primo passaggio

Dopo l'oracle:

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} |x\rangle |-\rangle .$$

Applichiamo $H^{\otimes n}$ al solo registro di ingresso:

$$\begin{aligned} |\psi_3\rangle &= (H^{\otimes n} \otimes I) |\psi_2\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} (H^{\otimes n} |x\rangle) |-\rangle . \end{aligned}$$

Le Hadamard finali: espansione della somma

Sostituiamo la formula per $H^{\otimes n}|x\rangle$:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} \left(\frac{1}{\sqrt{2^n}} \sum_y (-1)^{x\cdot y} |y\rangle \right) |-\rangle$$

$$= \frac{1}{2^n} \sum_x \sum_y (-1)^{f(x)+x\cdot y} |y\rangle |-\rangle .$$

I due fattori $1/\sqrt{2^n}$ danno $1/2^n$; i segni si moltiplicano e gli esponenti si sommano.

Raggruppiamo i termini per uscita y

La somma precedente può essere riordinata raccogliendo tutti i contributi allo stesso stato $|y\rangle$:

$$|\psi_3\rangle = \frac{1}{2^n}\sum_x\sum_y (-1)^{f(x)+x\cdot y}\,|y\rangle\,|-\rangle$$

$$= \sum_y \underbrace{\left[\frac{1}{2^n}\sum_x (-1)^{f(x)+x\cdot y}\right]}_{\alpha_y} |y\rangle\,|-\rangle\,.$$

$$\boxed{\alpha_y = \frac{1}{2^n}\sum_x (-1)^{f(x)+x\cdot y}.}$$

Perché basta esaminare lo stato $|0^n\rangle$?

Se l'uscita del registro è $y = 0^n$, per qualunque ingresso x:

$$x \cdot 0^n = 0.$$

L'ampiezza corrispondente diventa

$$\boxed{\alpha_{0^n} = \frac{1}{2^n} \sum_x (-1)^{f(x)}.}$$

Questa somma dipende solo da **quante volte** f vale 0 o 1, non dalla posizione degli input.

La promessa è dunque sufficiente per calcolare α_{0^n}.

Caso 1: funzione costante

Se $f(x) = 0$ **per ogni** x**:**

$$\alpha_{0^n} = \frac{1 + \cdots + 1}{2^n} = \frac{2^n}{2^n} = 1.$$

Se $f(x) = 1$ **per ogni** x**:**

$$\alpha_{0^n} = \frac{-1 - \cdots - 1}{2^n} = -\frac{2^n}{2^n} = -1.$$

In entrambi i casi:

$$\boxed{P(0^n) = |\alpha_{0^n}|^2 = 1.}$$

La misura dei n qubit restituisce sempre la stringa 0^n.

Caso 2: funzione bilanciata

Nella somma $\sum_x (-1)^{f(x)}$ compaiono esattamente:

- 2^{n-1} termini uguali a $+1$ (quando $f(x) = 0$);
- 2^{n-1} termini uguali a -1 (quando $f(x) = 1$).

Pertanto

$$\alpha_{0^n} = \frac{2^{n-1} - 2^{n-1}}{2^n} = 0.$$

$$\boxed{P(0^n) = |\alpha_{0^n}|^2 = 0.}$$

La misura **non può** restituire la stringa 0^n.

La regola di decisione

Misuriamo tutti i n qubit del registro di ingresso nella base computazionale.

Risultato della misura	Conclusione
$0^n = 00\cdots0$	funzione **costante**
$y \neq 0^n$	funzione **bilanciata**

$$P(0^n) = \begin{cases} 1, & f \text{ costante}, \\ 0, & f \text{ bilanciata}. \end{cases}$$

L'algoritmo è **esatto** nel modello ideale e utilizza una sola query.

Esempio $n = 3$: funzione costante

Scegliamo $f(x) = 1$ per tutti gli otto input. Il primo registro, dopo l'oracle, è

$$|\phi_2\rangle = -\frac{1}{\sqrt{8}} \sum_x |x\rangle = -H^{\otimes 3} |000\rangle .$$

Applichiamo $H^{\otimes 3}$, usando $H^2 = I$:

$$\begin{aligned} H^{\otimes 3} |\phi_2\rangle &= -H^{\otimes 3} H^{\otimes 3} |000\rangle \\ &= -|000\rangle . \end{aligned}$$

La fase globale $-$ non modifica la misura: $\boxed{\mathrm{P}(000){=}1.}$

Esempio $n = 2$: funzione bilanciata semplice

Consideriamo $f(x_1x_2) = x_1$:

x	00	01	10	11
$f(x)$	0	0	1	1

Dopo l'oracle, il primo registro è

$$|\phi_2\rangle = \frac{|00\rangle + |01\rangle - |10\rangle - |11\rangle}{2}$$
$$= \frac{|0\rangle - |1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}} = |-\rangle\,|+\rangle\,.$$

Esempio $n = 2$: la misura conclusiva

Dalla slide precedente, il registro di ingresso è $|-\rangle\,|+\rangle$. Le Hadamard finali danno

$$\begin{aligned}(H \otimes H)\,|-\rangle\,|+\rangle &= (H\,|-\rangle) \otimes (H\,|+\rangle) \\ &= |1\rangle \otimes |0\rangle = |10\rangle\,.\end{aligned}$$

La misura restituisce 10 con probabilità uno. Poiché $10 \neq 00$, la conclusione è **bilanciata**.

Una funzione bilanciata con più possibili uscite

Per $n = 3$, consideriamo la seguente funzione (dalla dispensa):

x	000	001	010	011	100	101	110	111
$f(x)$	1	1	1	0	1	0	0	0

Quattro uscite della funzione valgono 0 e quattro valgono 1. Dopo l'oracle, il registro ha lo stato

$$|\phi_2\rangle = \frac{1}{\sqrt{8}}(-|000\rangle - |001\rangle - |010\rangle + |011\rangle - |100\rangle + |101\rangle + |110\rangle + |111\rangle).$$

La somma delle fasi nell'esempio $n = 3$

Per ogni uscita y vale $\alpha_y = \frac{1}{8}\sum_x (-1)^{f(x)+x\cdot y}$.

y	$8\alpha_y$	α_y	y	$8\alpha_y$	α_y
000	0	0	100	-4	$-1/2$
001	-4	$-1/2$	101	0	0
010	-4	$-1/2$	110	0	0
011	0	0	111	4	$1/2$

Le quattro ampiezze non nulle hanno modulo $1/2$; quella di 000 è nulla.

L'esempio $n = 3$: stato e probabilità finali

Le Hadamard finali producono

$$\boxed{|\phi_3\rangle = \frac{1}{2}(-|001\rangle - |010\rangle - |100\rangle + |111\rangle).}$$

Le misure possibili sono

$$001, \quad 010, \quad 100, \quad 111$$

con probabilità $1/4$ ciascuna; tutte le altre hanno probabilità zero. **Non è necessaria un'uscita univoca:** è sufficiente che 000 non compaia mai. Questa proprietà permette di riconoscere la funzione bilanciata con certezza.

Perché il metodo funziona: interferenza

Dopo l'oracle, i valori $f(x)$ sono codificati nei segni $(-1)^{f(x)}$. Le Hadamard finali fanno interferire le ampiezze. Per l'uscita 0^n:

$$\alpha_{0^n} = \frac{1}{2^n} \sum_x (-1)^{f(x)}.$$

Costante: tutti i contributi hanno lo stesso segno, quindi $|\alpha_{0^n}| = 1$. **Bilanciata:** i contributi positivi e negativi si cancellano, quindi $\alpha_{0^n} = 0$.

Confronto delle query: quale vantaggio?

Nel modello a oracle, imponendo una risposta **esatta**:

$$\boxed{Q_{\mathrm{C}}^{\mathrm{det,exact}}(n) = 2^{n-1} + 1, \qquad Q_{\mathrm{Q}}^{\mathrm{exact}}(n) = 1.}$$

n	Query classiche (caso peggiore)	Query quantistiche
1	2	1
3	5	1
10	513	1

La separazione cresce esponenzialmente con n nelle *query*. Non si tratta automaticamente di un vantaggio equivalente nel tempo fisico.

Il confronto ha un'ipotesi importante

Il limite $2^{n-1}+1$ riguarda algoritmi **classici deterministici** che devono rispondere senza errore. Se ammettiamo una piccola probabilità di errore, un algoritmo classico può

campionare alcuni input casuali:

- se osserva entrambi i valori, conclude che f è bilanciata;
- se osserva sempre lo stesso valore, può dichiarare "costante" con un rischio controllato.

Il numero di query classiche probabilistiche con errore limitato può essere $O(1)$, indipendente da n. Il vantaggio esponenziale riguarda dunque il **modello esatto**

deterministico.

Deutsch è il caso $n = 1$

Per $n = 1$ il dominio contiene soltanto 0 e 1; il problema coincide con quello della Lect18. La formula per l'ampiezza dell'uscita zero è

$$\alpha_0 = \frac{1}{2}\left[(-1)^{f(0)} + (-1)^{f(1)}\right].$$

- Se $f(0) = f(1)$, allora $|\alpha_0| = 1$ e la misura restituisce 0.
- Se $f(0) \neq f(1)$, allora $\alpha_0 = 0$ e la misura restituisce 1.

Deutsch–Jozsa generalizza lo stesso meccanismo a n qubit di ingresso.

Applicazione: collaudo logico con promessa

Un circuito di controllo con n ingressi digitali produce un segnale di allarme $f(x)$. La specifica garantisce **una** delle due possibilità:

- l'allarme è identico per tutte le 2^n configurazioni;
- l'allarme è attivo per esattamente metà delle configurazioni.

Il test deve classificare il circuito come **costante** o **bilanciato**. Nel modello a oracle quantistico, Deutsch–Jozsa svolge il test con una query esatta.

Conclusioni

1. Una sovrapposizione uniforme contiene tutti i 2^n ingressi.
2. Una sola query all'oracle trasferisce $f(x)$ nei segni delle ampiezze.
3. Le Hadamard finali trasformano tali segni in un criterio di misura esatto:

$$P(0^n) = 1 \text{ se costante}, \qquad P(0^n) = 0 \text{ se bilanciata}.$$

$$\boxed{Q_{\mathrm{Q}}^{\mathrm{exact}}(n) = 1.}$$