Quantum Information Processing:

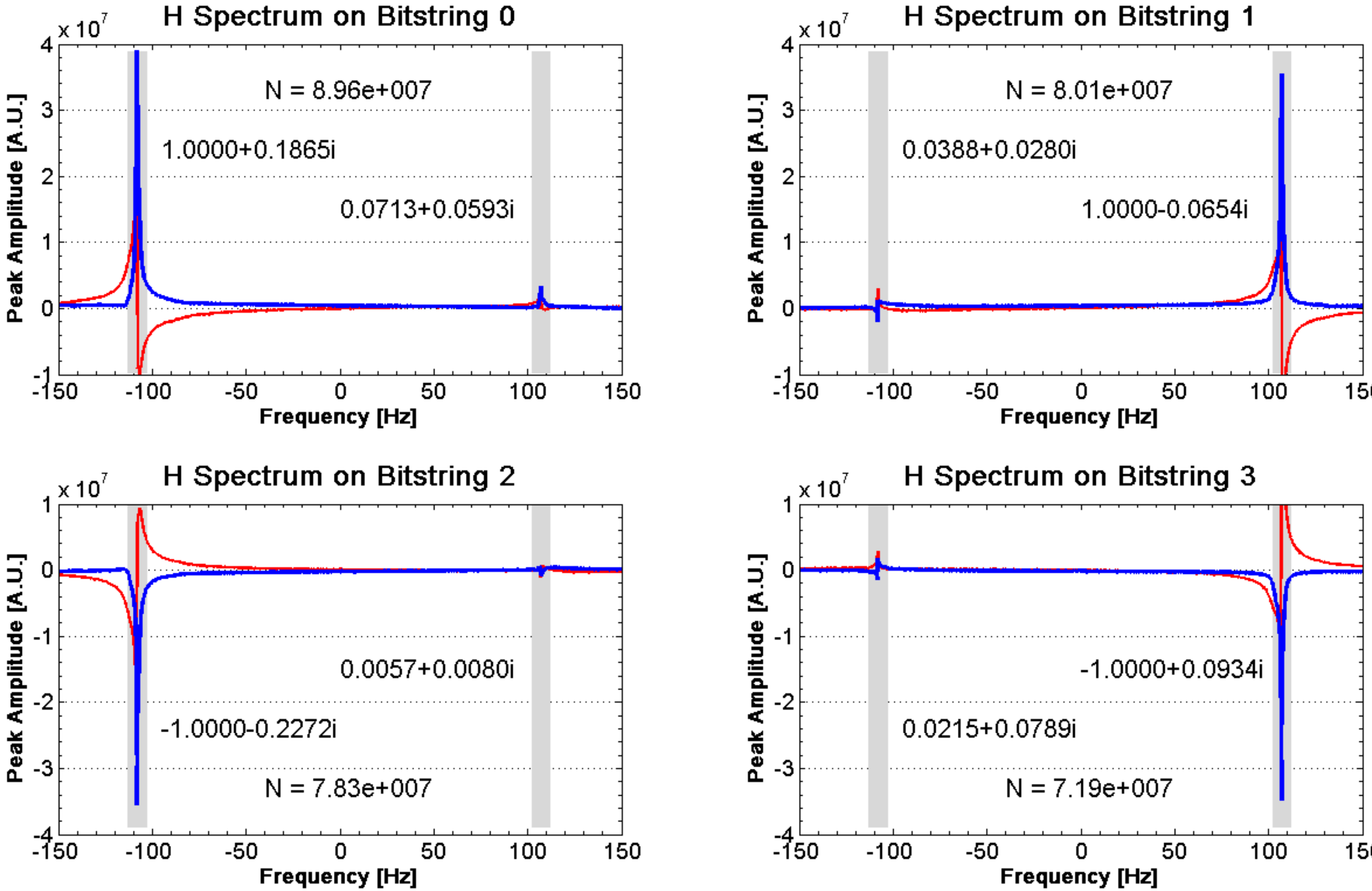
Deutsch Algorithm and Grover Search

Edwin Ng | 2 May 2012

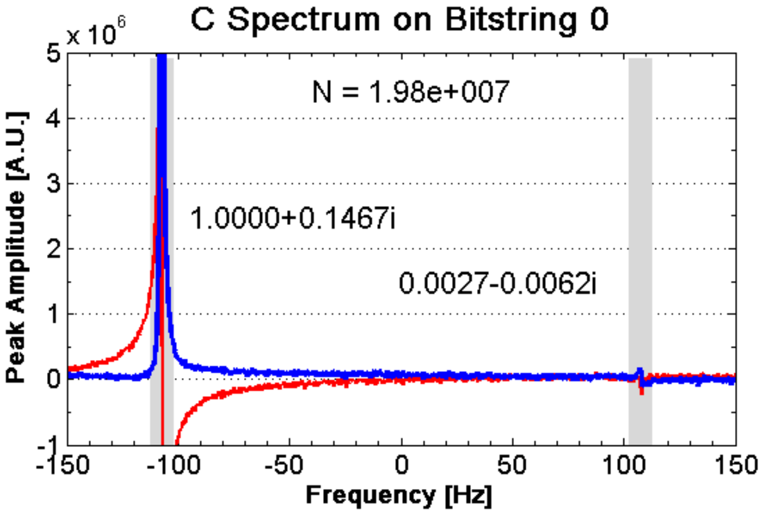
The Computational Basis

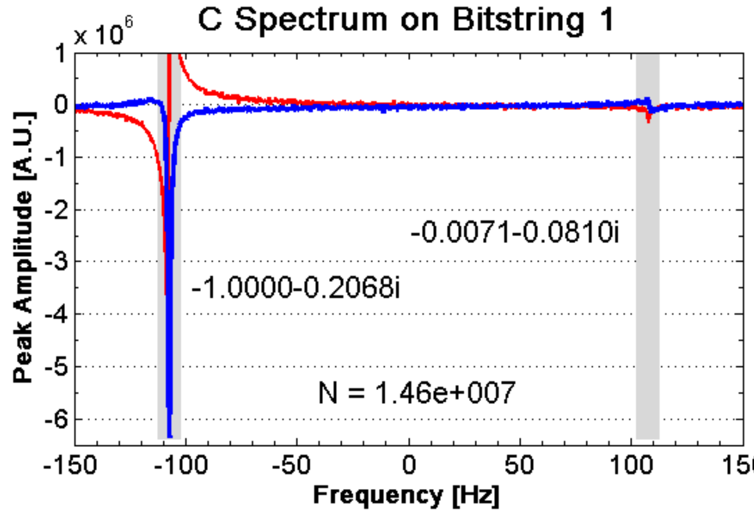
- The computational basis states of the molecule are $|00\rangle, |01\rangle, |10\rangle, |11\rangle$
- These correspond to the *classical bits*
- NMR quantum computation manipulates superpositions of these basis states to solve problems faster than classical algorithms

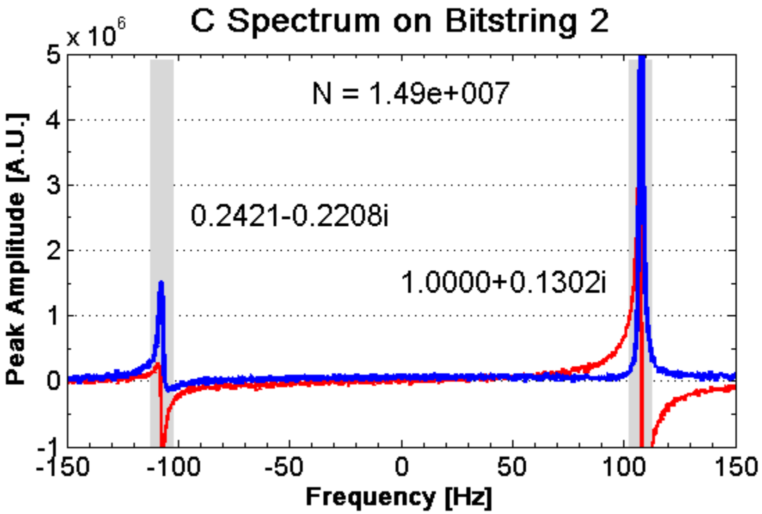
The Computational Basis: FIDs

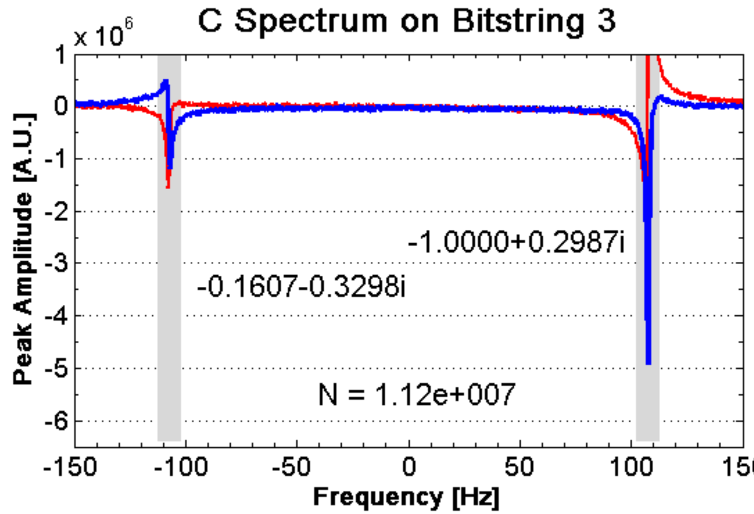


The Computational Basis: FIDs









Basis Probabilities

- If the state is measured in the computational basis, what is the probability of each state?
- After normalization, the proton and carbon FIDs gives V_1^H, V_2^H, V_1^C, V_2^C
- They represent the following system:

$$V_1^H = (\rho_{11} - \rho_{33}) - i\,(\rho_{31} + \rho_{13})$$
$$V_2^H = (\rho_{22} - \rho_{44}) - i\,(\rho_{24} + \rho_{42})$$
$$V_1^C = (\rho_{11} - \rho_{22}) - i\,(\rho_{21} + \rho_{12})$$
$$V_2^C = (\rho_{33} - \rho_{44}) - i\,(\rho_{43} + \rho_{34})$$

Basis Probabilities (Cont.)

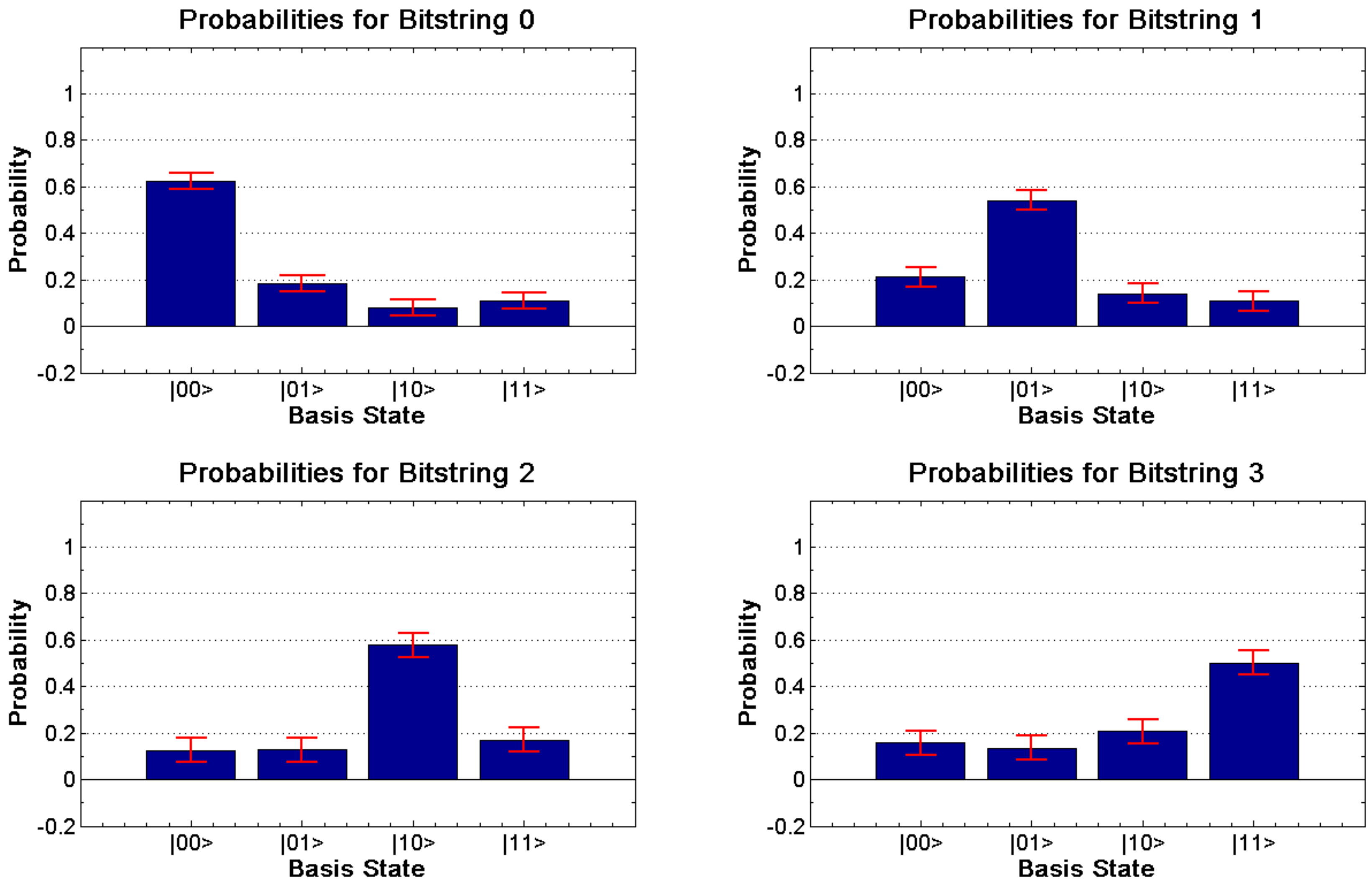
- ρ_{jj} represents probability of measuring the *j*-th basis element

- We do not need the imaginary elements

- System is rank-deficient: add normalization

$$\rho_{11} + \rho_{22} + \rho_{33} + \rho_{44} = 1$$

Basis Probabilities (Cont.)



Simple Quantum Gates: One-Qubit

- NMR is based on single-qubit *rotation* gates:

$$X = \exp\left(-i\frac{\pi}{4}\sigma_x\right), \quad Y = \exp\left(-i\frac{\pi}{4}\sigma_y\right)$$

- These rotate the spin by π/2 about *x*, *y* axis of the NMR system (π/2 pulses).

- X^2 and Y^2 are π pulses; we also have -π/2

$$\overline{X} = X^\dagger, \quad \overline{Y} = Y^\dagger$$

Simple Quantum Gates: Two-Qubits

- In two-qubit NMR, the two nuclei couple together through J-coupling constant

- This yields spin-spin interaction operator

$$U_\tau = \exp\left(-i\frac{\pi}{4}\sigma_z \otimes \sigma_z\right)$$

- Achieved by letting system freely evolve for time $\tau = 1/2J$

The Controlled-NOT (CNOT) Gate

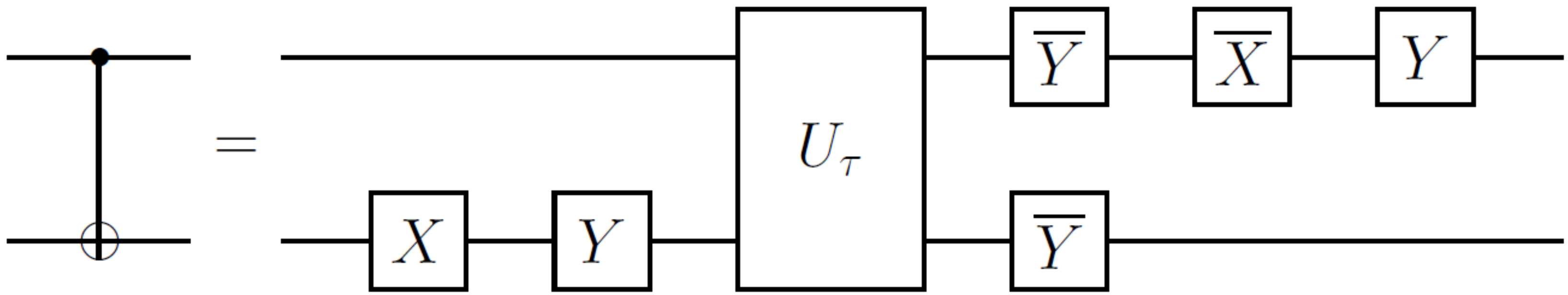
- Defined by $C|i\rangle|j\rangle = |i\rangle|i \oplus j\rangle$
- Classical Truth Table:

b	00	01	10	11
$C(b)$	00	01	11	10

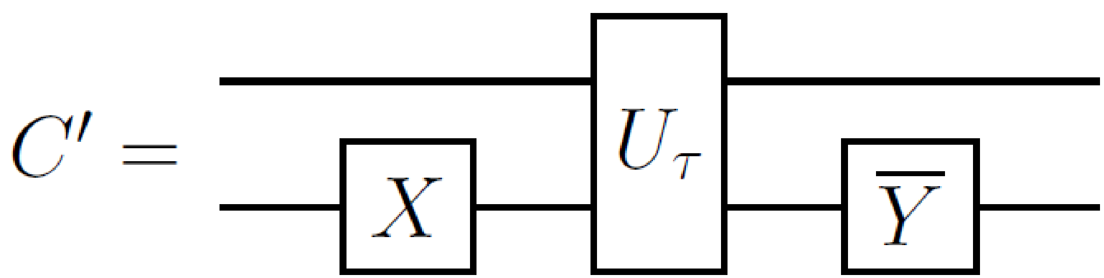
- The first bit is the control, the second bit is the target. CNOT flips target iff control is 1.

The CNOT Gate: Circuit

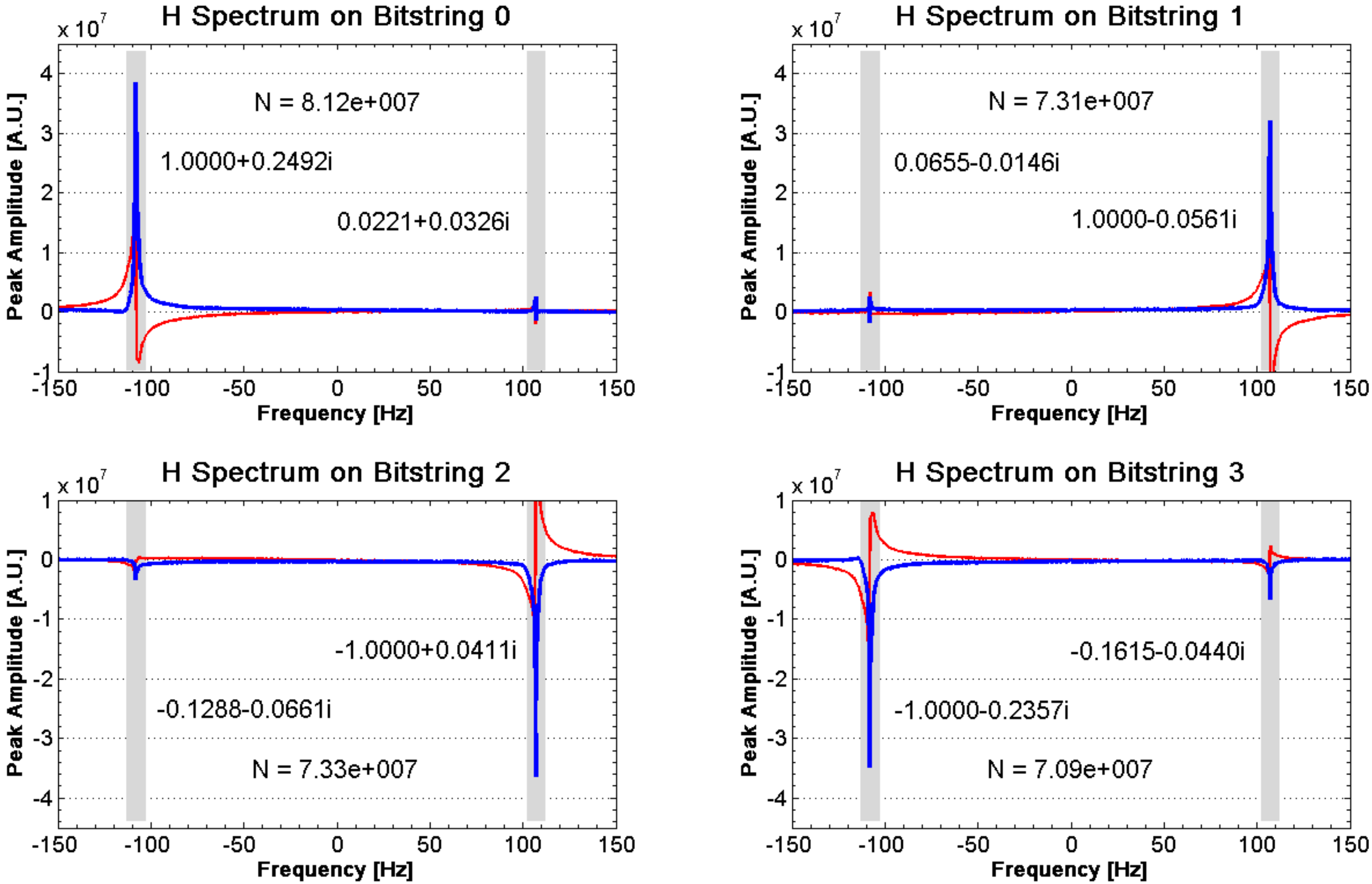
- Quantum CNOT is a two qubit-circuit



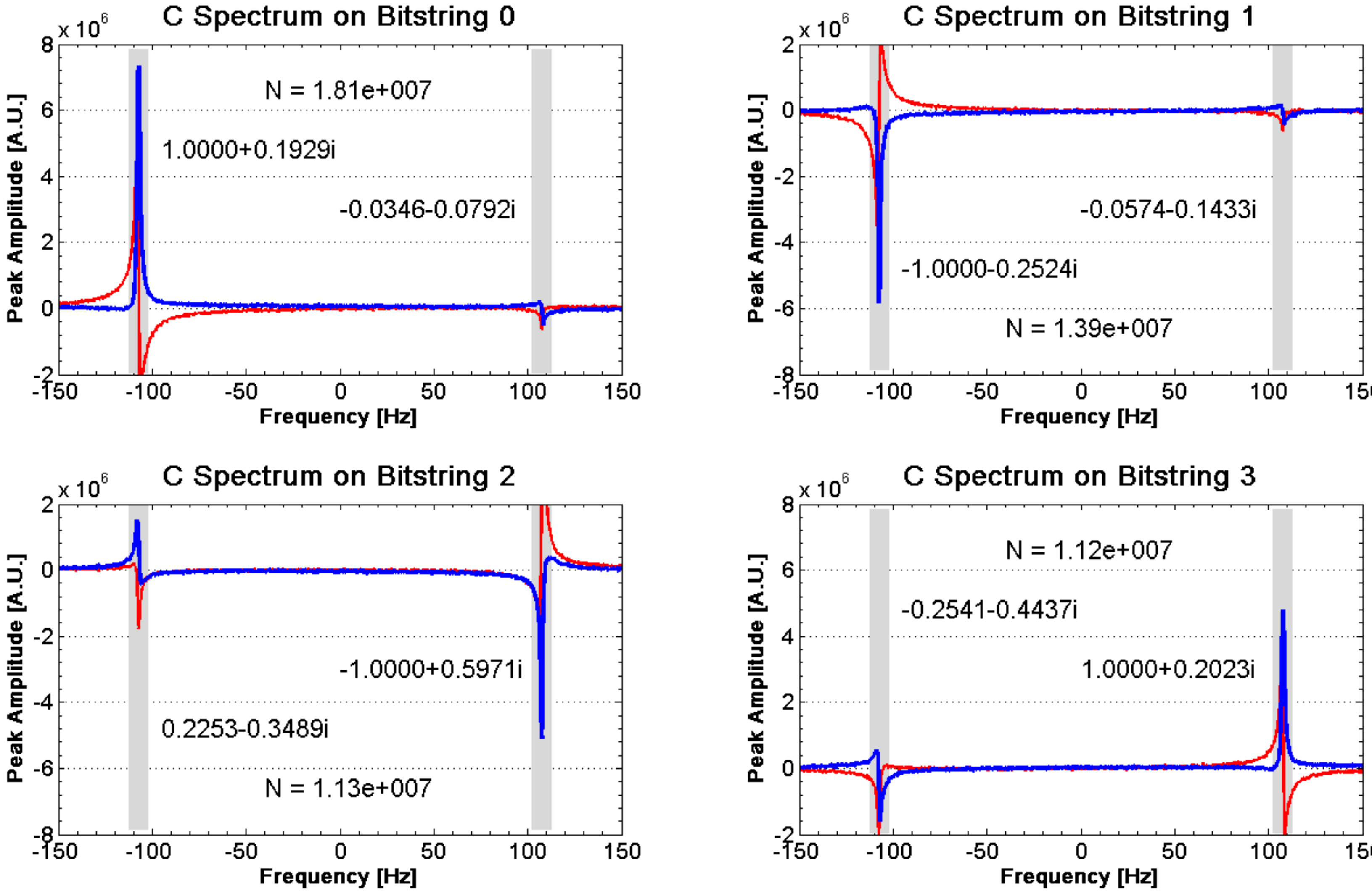
- There is also a much simpler near-CNOT gate, disregarding phases



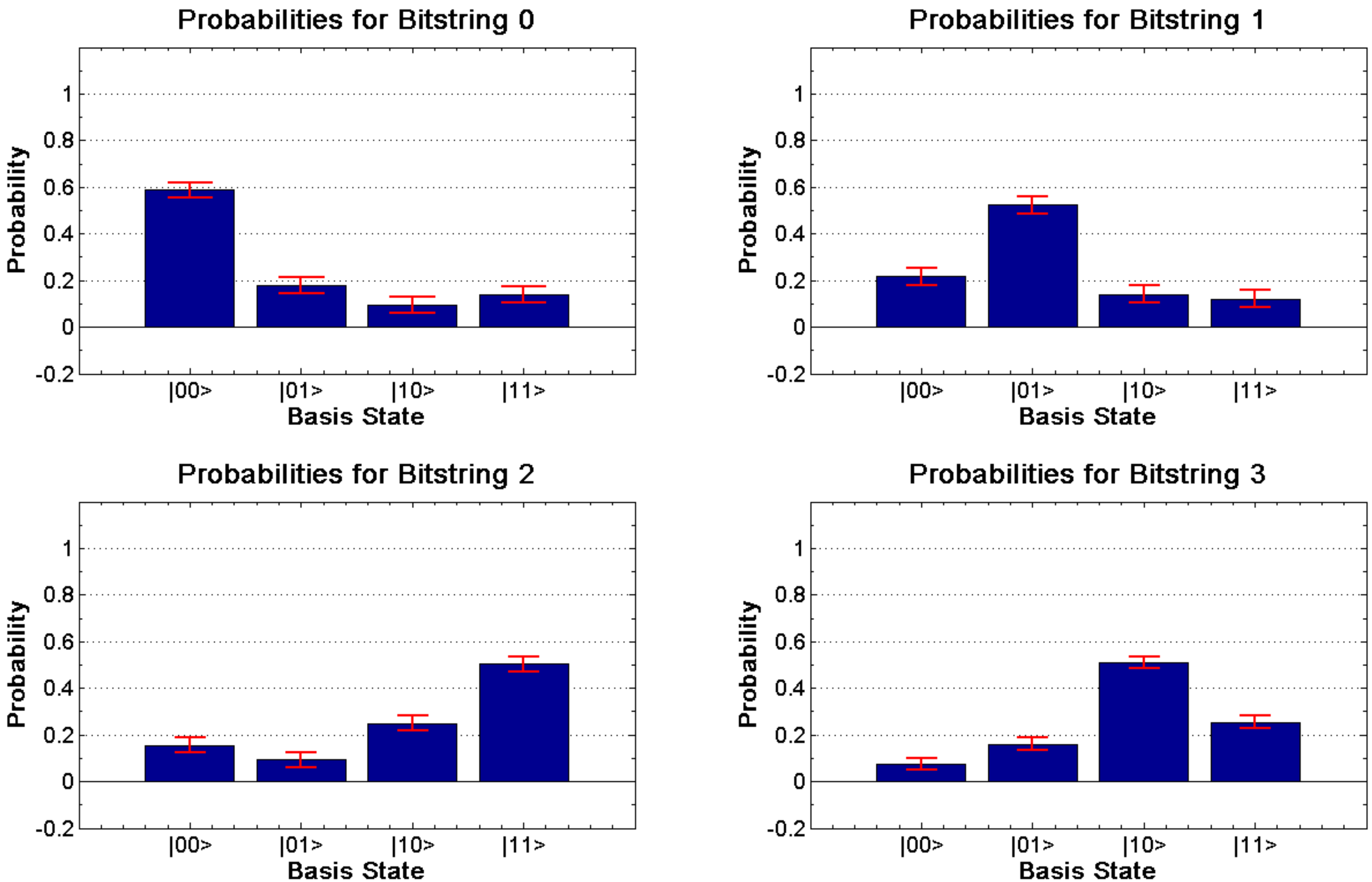
The CNOT Gate: FIDs



The CNOT Gate: FIDs



The CNOT Gate: Probabilities



The Deutsch Algorithm: Question

- Given a function $f : \{0, 1\} \to \{0, 1\}$

- Constant
  - f_0 and f_3

  vs.

- Balanced
  - f_1 and f_2

b	$f_0(b)$
0	0
1	0

b	$f_1(b)$
0	0
1	1

b	$f_2(b)$
0	1
1	0

b	$f_3(b)$
0	1
1	1

The Deutsch Algorithm: Setup

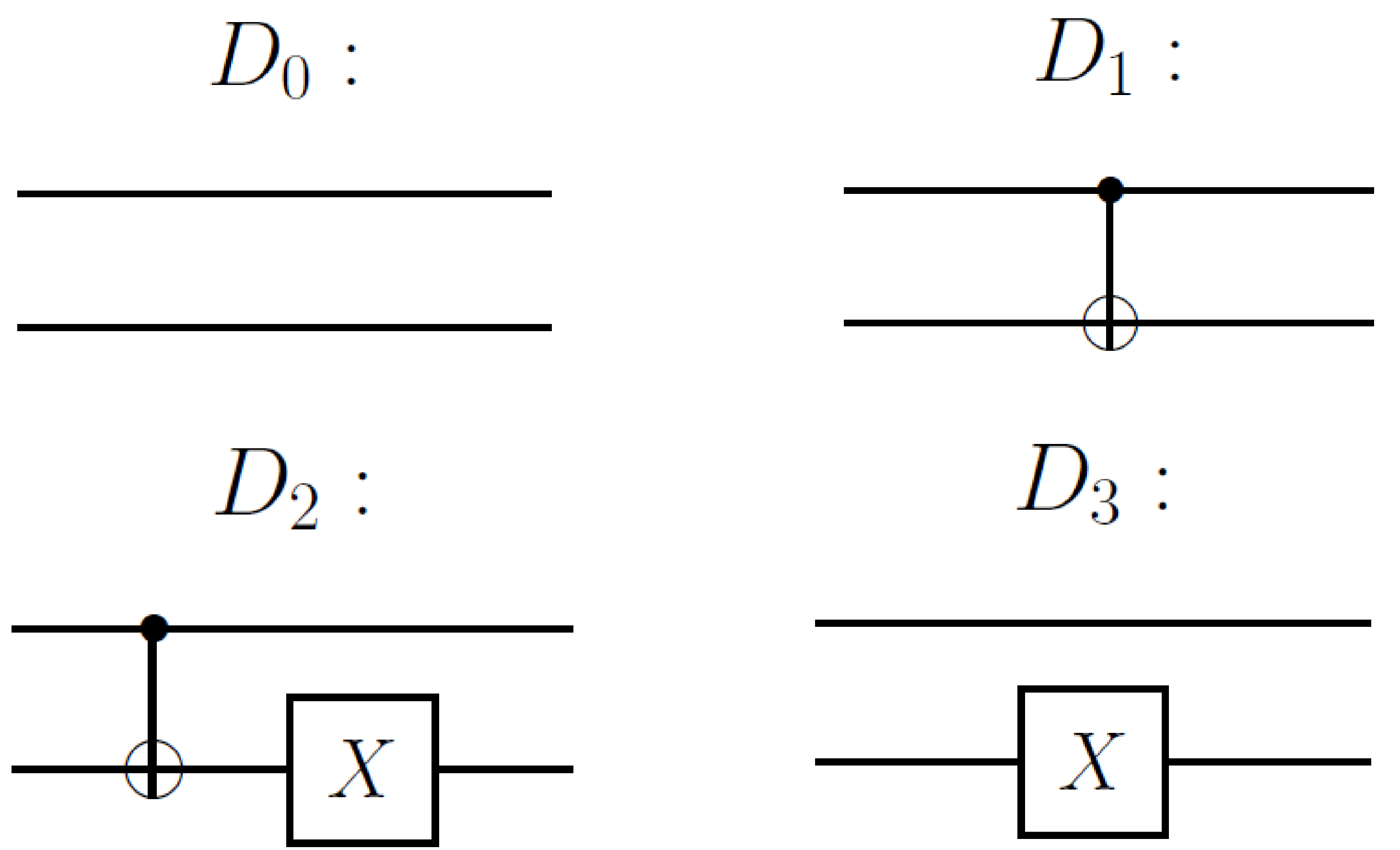
- *Classical approach*: Ask for both *f*(0) and *f*(1)
- *Quantum approach*: Ask for only one thing, but need to choose that one thing carefully

$$D|b_1\rangle|b_2\rangle = |b_1\rangle|f(b_1) \oplus b_2\rangle$$

- *D* is a unitary operator: *i.e.*, a quantum gate
- Goal is to query *D* at most one time, which would beat classical case

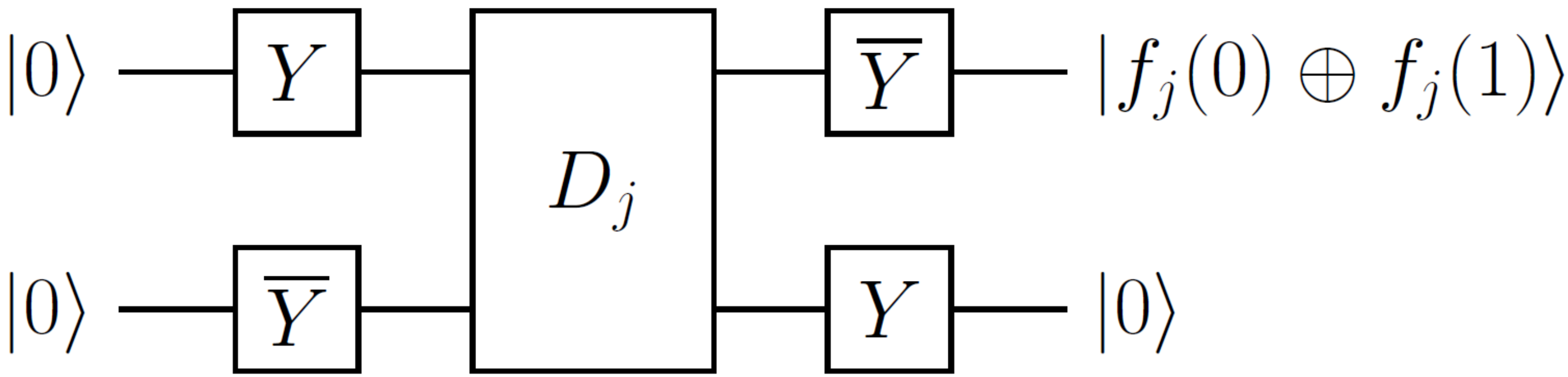
The Deutsch Algorithm: Setup (Cont.)

- For each f_j, there is a D_j *oracle*



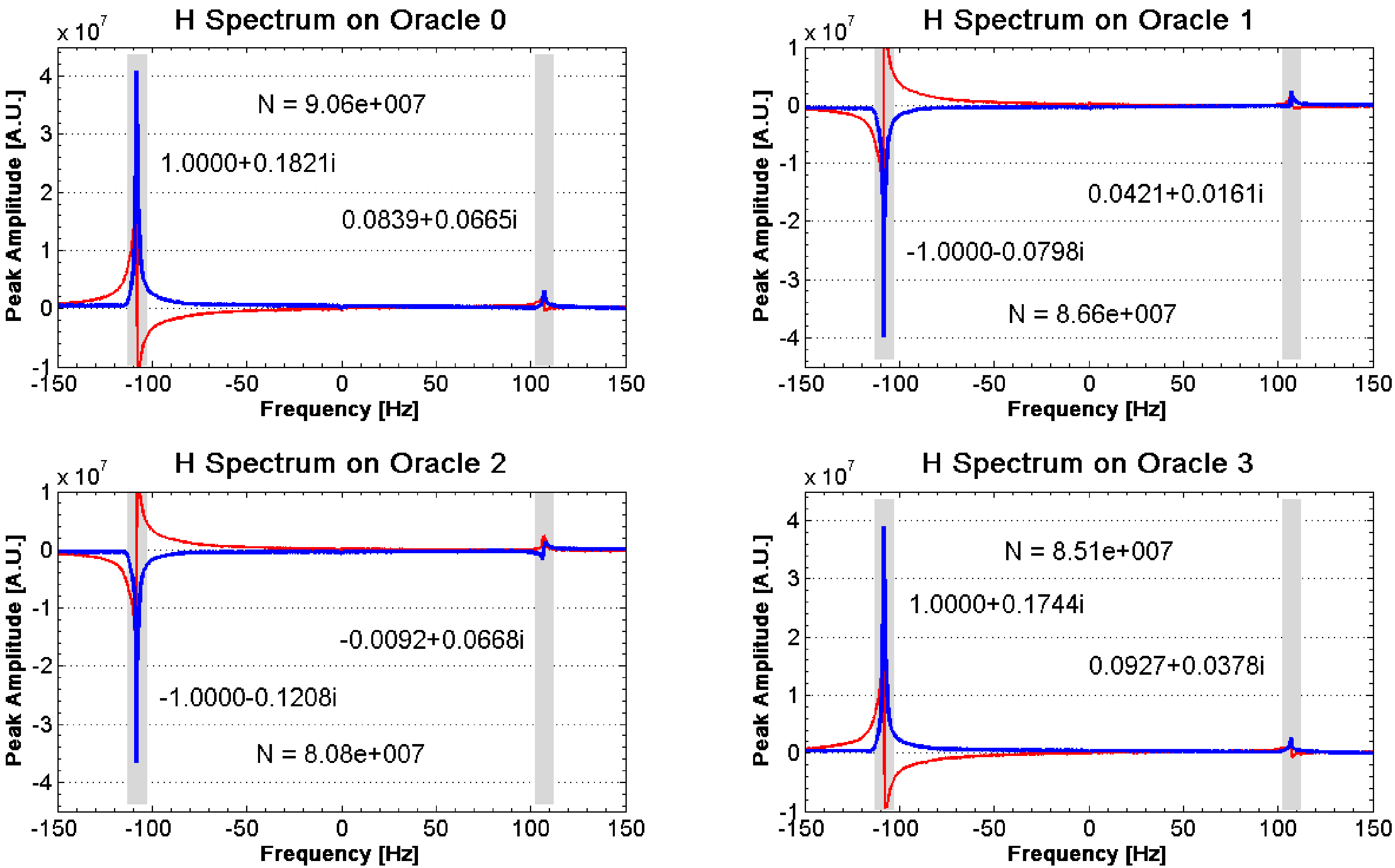
The Deutsch Algorithm: Solution

- The following quantum circuit solves the Deutsch problem in one query of *D*:

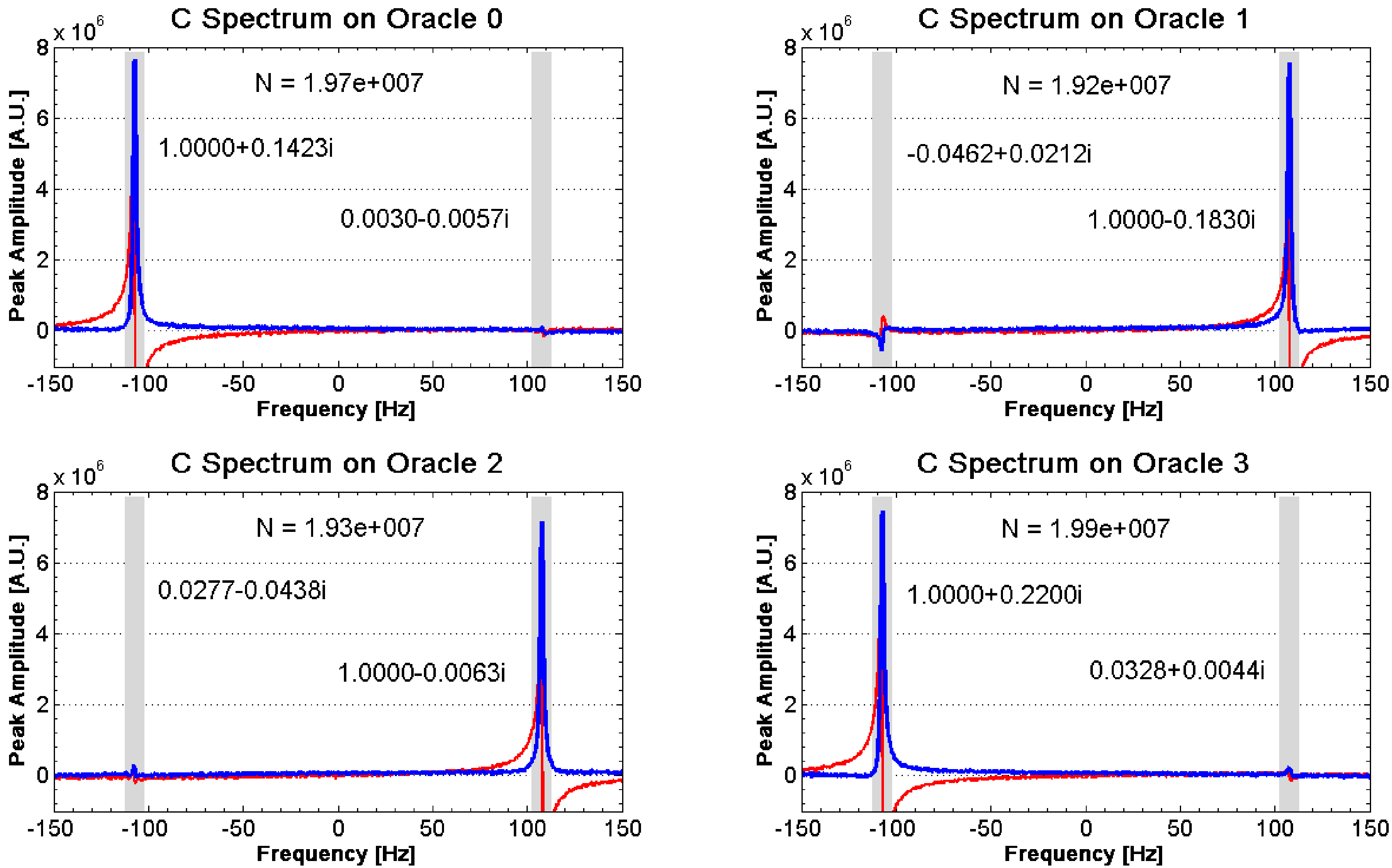


- Measuring gives 00 if constant, 10 if balanced

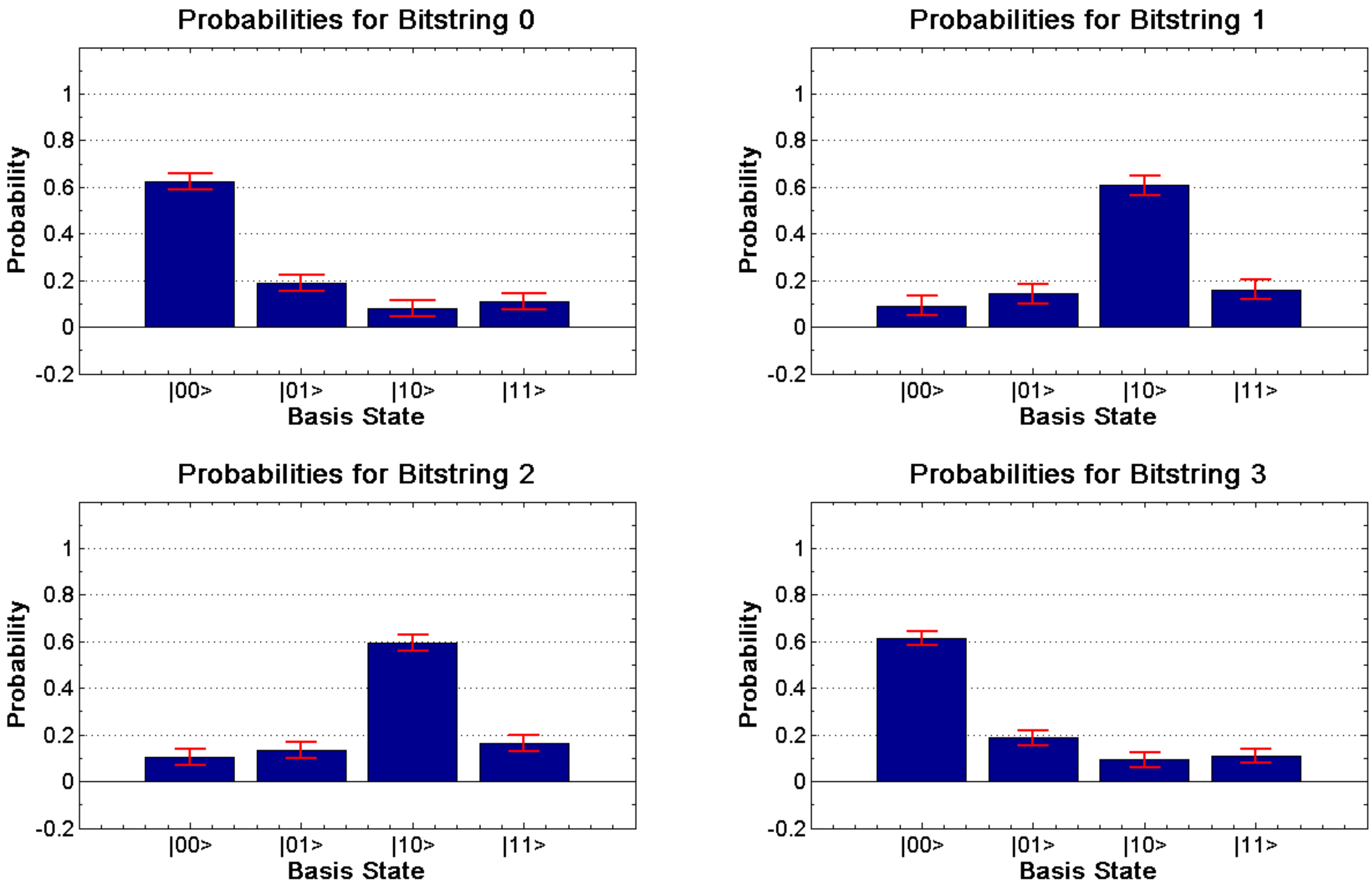
The Deutsch Algorithm: FIDs



The Deutsch Algorithm: FIDs



The Deutsch Algorithm: Probabilities



The Grover Algorithm: Question

- Given a set *X* of *N* items and $g : X \to \{0, 1\}$
- Exactly one element x_0 is marked 1
- *Goal*: Find x_0

- Classical approach is to just search all of *X*
  - This takes time $O(N)$
- Quantum approach indexes *X* using states
  - Ultimately takes time $O(\sqrt{N})$

The Grover Algorithm: Setup

- Instead of querying *g*, ask for an oracle instead
- *O* is a unitary operator on basis bitstrings *x*:

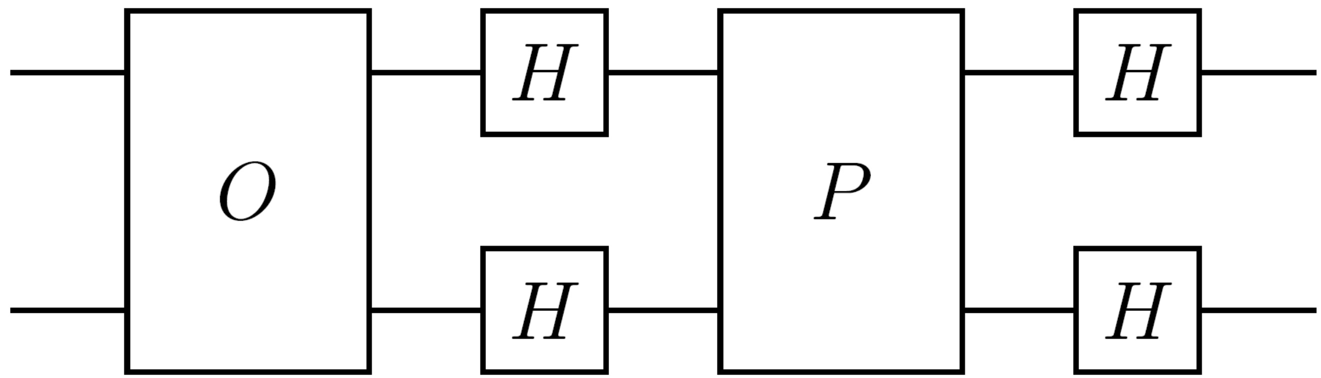
$$O|x\rangle = (-1)^{g(x)}|x\rangle$$

- Marks the answer using a "phase kickback"
- How to phrase the oracle query?

The Grover Algorithm: Setup (Cont.)

- A single query consists of the Grover iterate

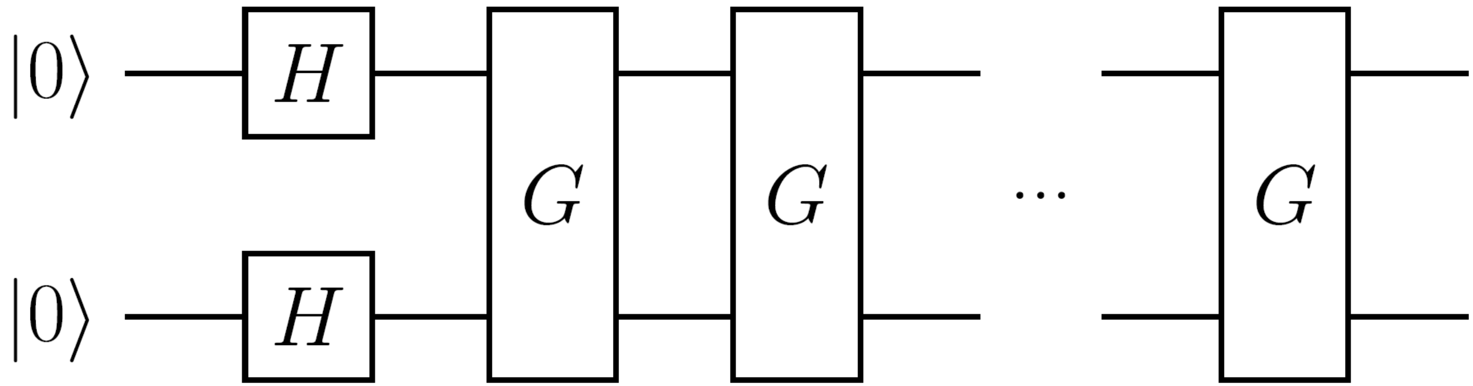
$G =$



- *P* is a conditional phase $P|x\rangle = (-1)^x|x\rangle$
- *H* is the Hadamard operator $\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

The Grover Algorithm: Solution

- Goal: Use as few Grover iterates as possible



- Measuring at the end of $k = O(\sqrt{N})$ iterations gives x_0 with high probability
- Will also get x_0 after $k+k_0$, $k+2k_0$, … iterations

The Grover Algorithm: Implementation

- Ignoring global phases and simplifying, we get a pulse sequence for each Grover iterate

$$G_0 = \overline{X}_1\overline{X}_2Y_1Y_2U_\tau\overline{X}_1\overline{X}_2Y_1Y_2U_\tau$$

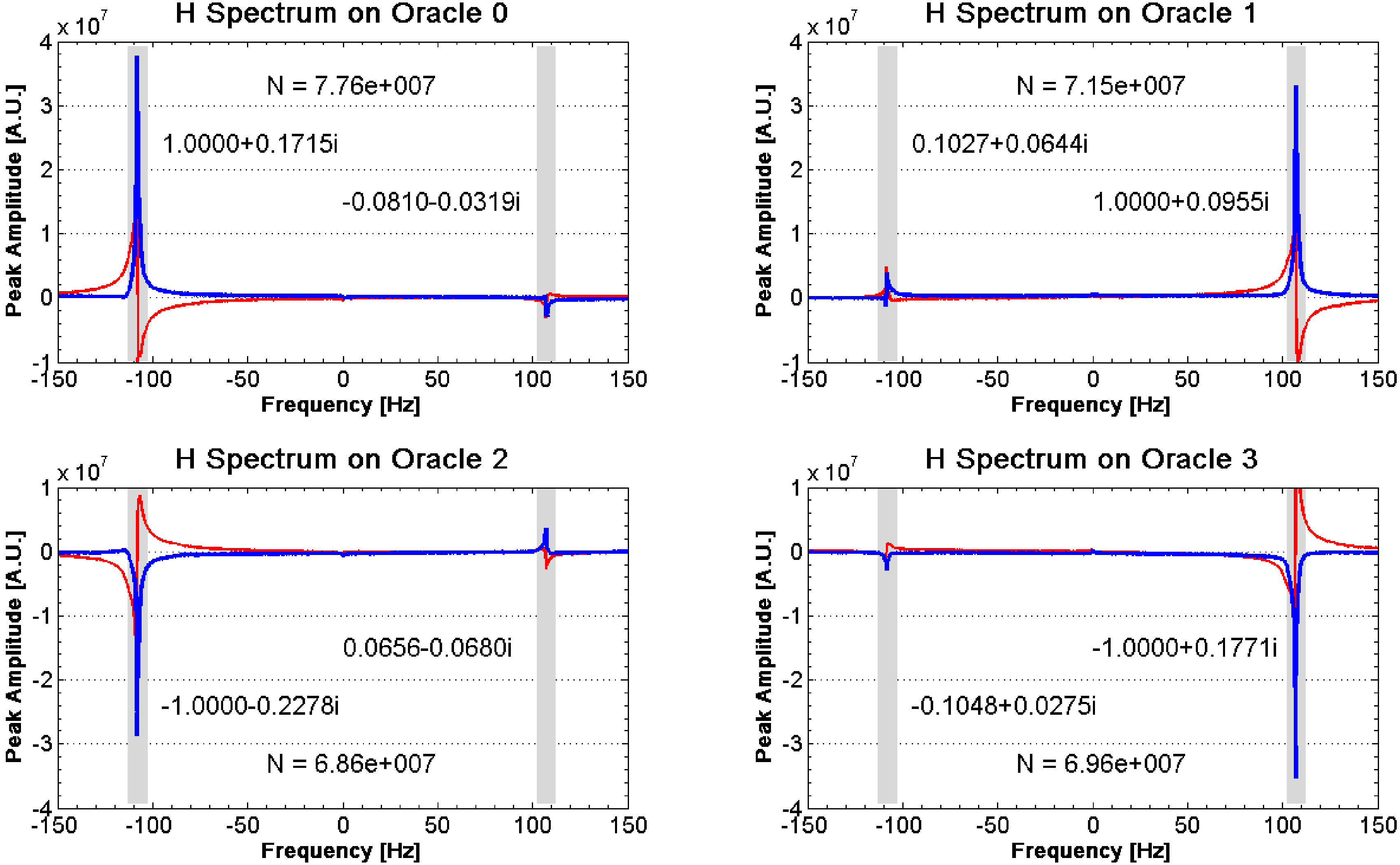
$$G_1 = \overline{X}_1\overline{X}_2Y_1Y_2U_\tau X_1\overline{X}_2Y_1Y_2U_\tau$$

$$G_2 = \overline{X}_1\overline{X}_2Y_1Y_2U_\tau\overline{X}_1X_2Y_1Y_2U_\tau$$

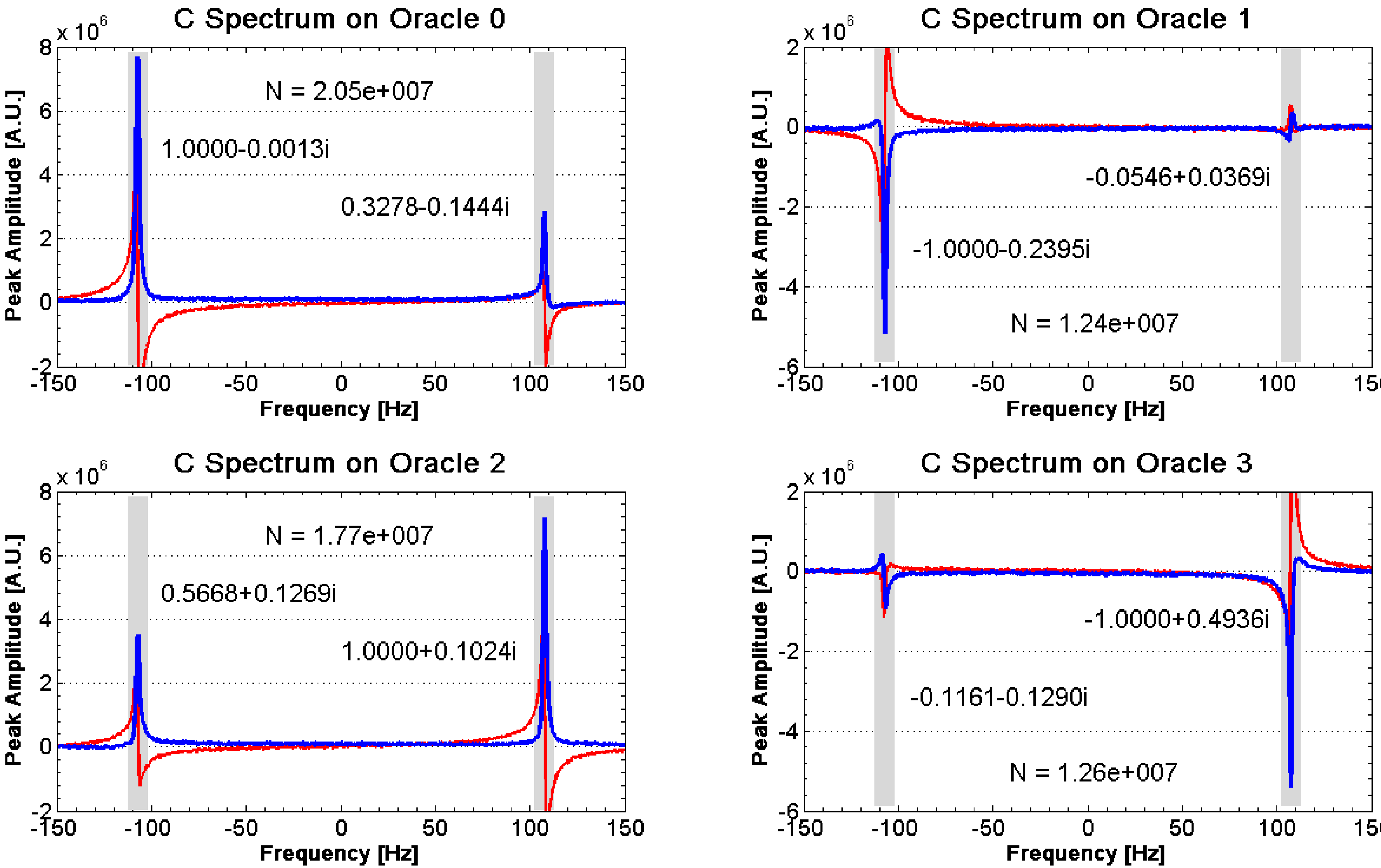
$$G_3 = \overline{X}_1\overline{X}_2Y_1Y_2U_\tau X_1X_2Y_1Y_2U_\tau$$

- The Hadamard is $H = YX^2$

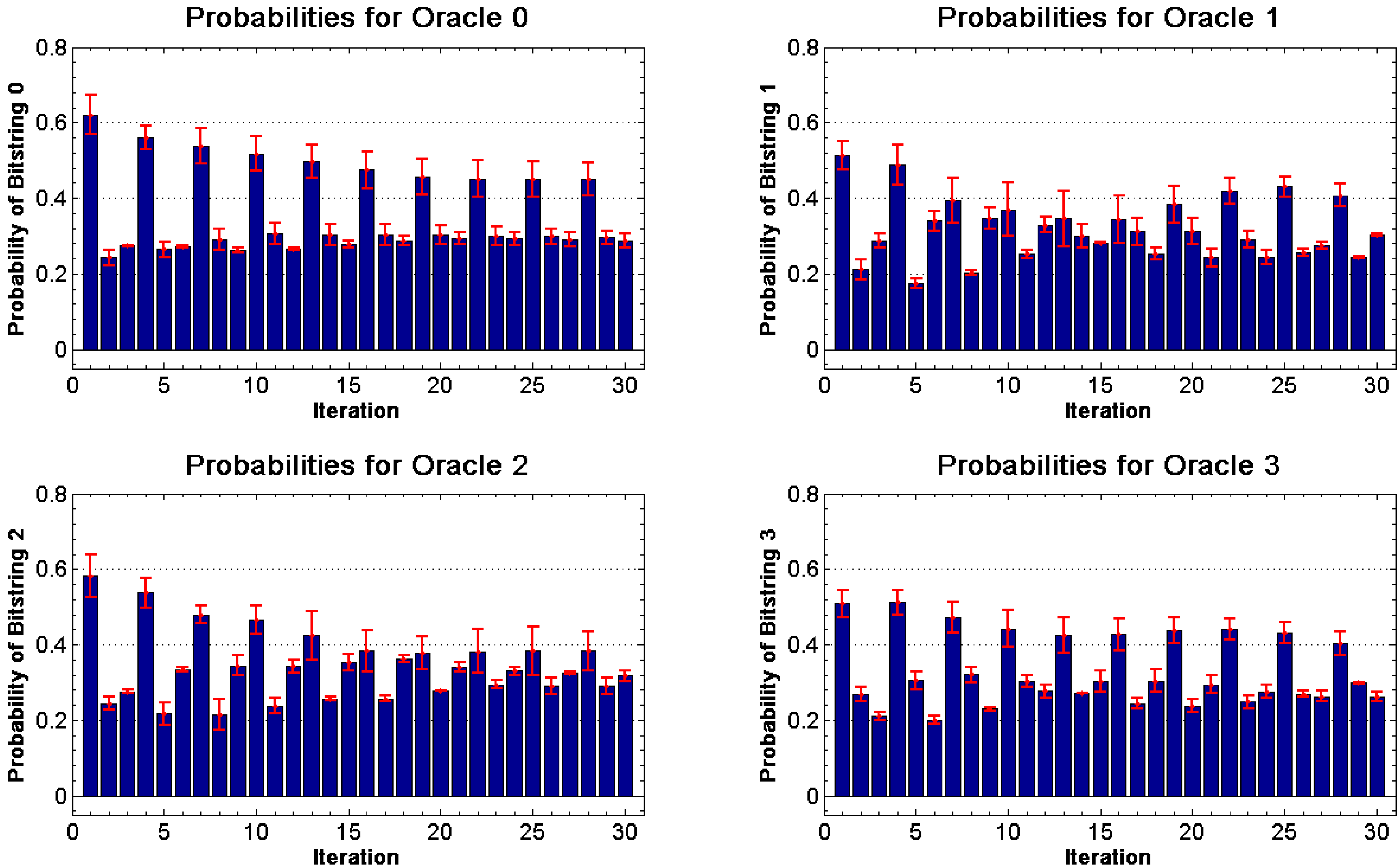
The Grover Algorithm: FIDs



The Grover Algorithm: FIDs



The Grover Algorithm: Probabilities



Conclusions

- Introduced a way to calculate the probabilities of each basis element after a computation
- Demonstrated the preparation of basis states
- Obtained a CNOT gate with correct classical outputs
- Verified the correctness of the Deutsch algorithm
- Observed the correctness and oscillatory behavior of the Grover search algorithm
- Also available:
  - Classical truth table for near-CNOT gate
  - Near-CNOT, CNOT, Deutsch using *carbon* control

Question and Answer