

Ch6: Benchmarking Quantum Computers

Basic idea

Benchmarking is how the performance of a computing system is determined.

One must choose the appropriate benchmark and metrics to extract meaningful results.

Given that QPUs have noise, how do you characterize what QPU you have ?

How do you tell if its good enough to be better than CPUs ?





Levels of benchmarking

One can distinguish three different levels of benchmarking:

- (0) – **Qubit benchmarking:** Benchmarking a single (or two) qubit system – how reliable the quantum gates are ?
- (1) – **Quantifying capabilities:** How much work a quantum computer is capable of, in contrast to performance results on specific algorithms.
- (2) – **Program benchmarks:** Establishing a set of programs and measuring the performance of a computing system performing each one.

Pure states

Fidelity is a measure of the “closeness” of two quantum states. It expresses the probability that one state will pass a test to identify as the other.

Fidelity as a distance measure between pure states is given by the transition probability from one state to another.

That is for two states described by unit vectors $|\psi\rangle, |\phi\rangle$:

$$F(|\phi\rangle, |\psi\rangle) = |\langle\phi|\psi\rangle|^2$$

Mixed states

For a pure state $|\psi\rangle$ and a mixed state ρ this generalizes to the averaged fidelity:

$$F(\rho, |\psi\rangle) = \langle\psi|\rho|\psi\rangle$$

And for two density matrices ρ, σ it is generalized as the largest fidelity between any two purifications of the given states.

According to Uhlmann's theorem, this leads to the expression:

$$F(\rho, \sigma) = \left[\text{Tr} \left\{ \sqrt{\sqrt{\rho}\sigma\sqrt{\rho}} \right\} \right]^2$$

Mixed states

If $[\rho, \sigma] = 0$, the definition simplifies to:

$$F(\rho, \sigma) = \left[\text{Tr}\{\sqrt{\rho\sigma}\} \right]^2 = \left(\sum_k \sqrt{p_k q_k} \right)^2$$

where p_k, q_k are the eigenvalues of ρ, σ respectively.

To see this, remember that if two operators commute then they can be diagonalized in the same basis:

$$\rho = \sum_k p_k |k\rangle\langle k| \qquad \sigma = \sum_k q_k |k\rangle\langle k|$$

so that:

$$\text{Tr}\{\sqrt{\rho\sigma}\} = \text{Tr}\left\{ \sum_k \sqrt{p_k q_k} |k\rangle\langle k| \right\} = \sum_k \sqrt{p_k q_k}$$

Properties

Other basic properties:

- Bounded: $0 \leq F(\sigma, \rho) \leq 1$ ($F(\sigma, \rho) = 1 \iff \sigma = \rho$)
- Symmetric: $F(\sigma, \rho) = F(\rho, \sigma)$
- Unitary invariant: $F(U\sigma U^\dagger, U\rho U^\dagger) = F(\sigma, \rho)$
- Monotonic: $F(\mathcal{E}(\sigma), \mathcal{E}(\rho)) \geq F(\sigma, \rho)$



1-qubit

So to measure fidelity we need to prepare states and measure their density matrix ρ . We must have multiple copies of ρ in order to characterize it.

Suppose we have multiple copies of ρ – or can make them on demand, we can expand:

$$\rho = \frac{\text{Tr}\{\rho\}I + \text{Tr}\{X\rho\}X + \text{Tr}\{Y\rho\}Y + \text{Tr}\{Z\rho\}Z}{2}$$

(I, X, Y, Z form a basis for 2×2 complex matrices). $\text{Tr}\{X\rho\}$ is $\langle X \rangle_\rho$, i.e. the expectation value of ρ when measured by the X observable – see Ch5.

Thus to get ρ we must calculate $\langle X \rangle_\rho, \langle Y \rangle_\rho, \langle Z \rangle_\rho$ (the 4th from normalization) \rightarrow 3 sets of code – see Ex. 3 assignment #1.



N-qubits

Quantum state tomography scales really badly ! For n -qubits you need $(2^{n^2} - 1)$ types of observables.

It works well but it takes time...



Quantum process tomography

Analogous to quantum state tomography except instead of learning a density matrix ρ we learn a representation of a quantum process, see Nielsen and Chuang chapter 8.4.2.

Standard QPT requires 2^{n^2} different state tomography experiments for n-qubits.

That is $2^{n^2} \times 2^{n^2}$ types of programs to benchmark n-qubits...

T1 coherence

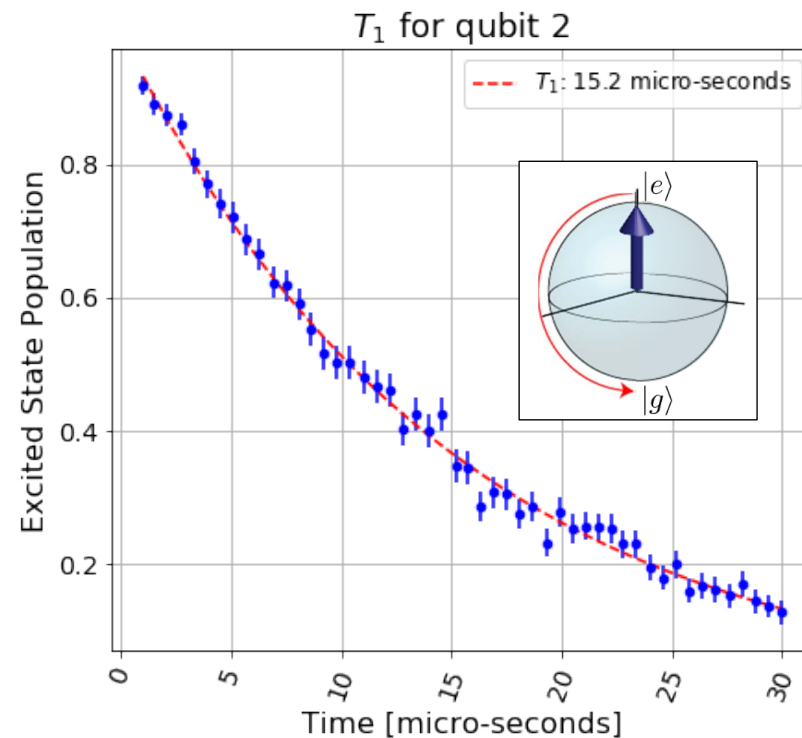
T1 measures the impact of amplitude damping – the loss of energy from the system.

Basic idea:

- Initialize the qubit to state 0
- Apply X
- Wait for time t
- Measure the probability of being in state 1

We expect an exponential decay:

$$P_1(t) = Ae^{-t/T_1} + C$$





T2 coherence

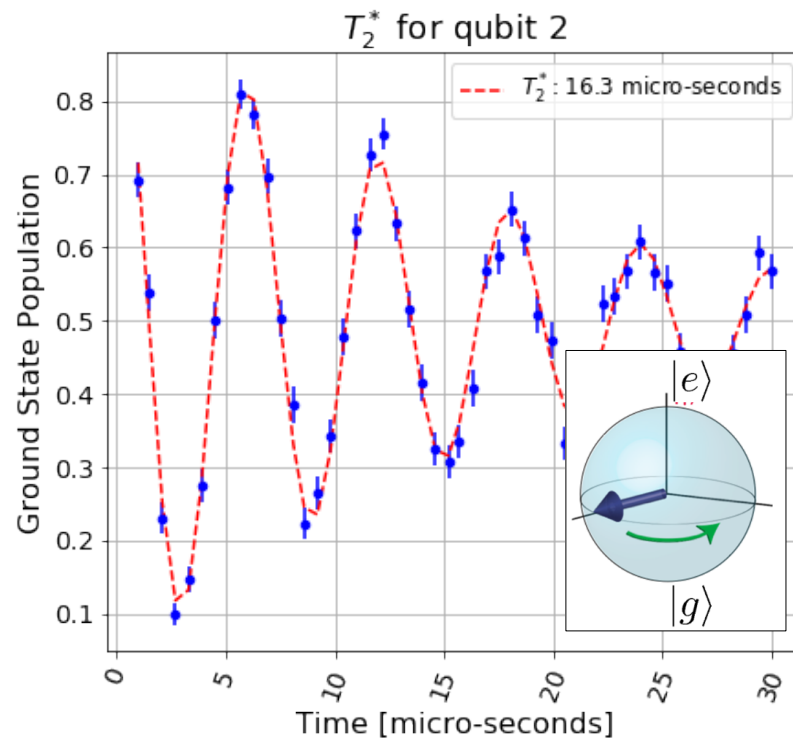
T2 measures the impact of phase damping – phase shifted version of T1.

Basic idea:

- Initialize the qubit to state 0
- Apply H
- Wait for time t
- Apply H
- Measure the probability of being in state 1

We expect a modulated exponential decay:

$$P_1(t) = Ae^{-2t/T_2} \sin(\omega_d(t - \phi)) + C$$





Basic idea

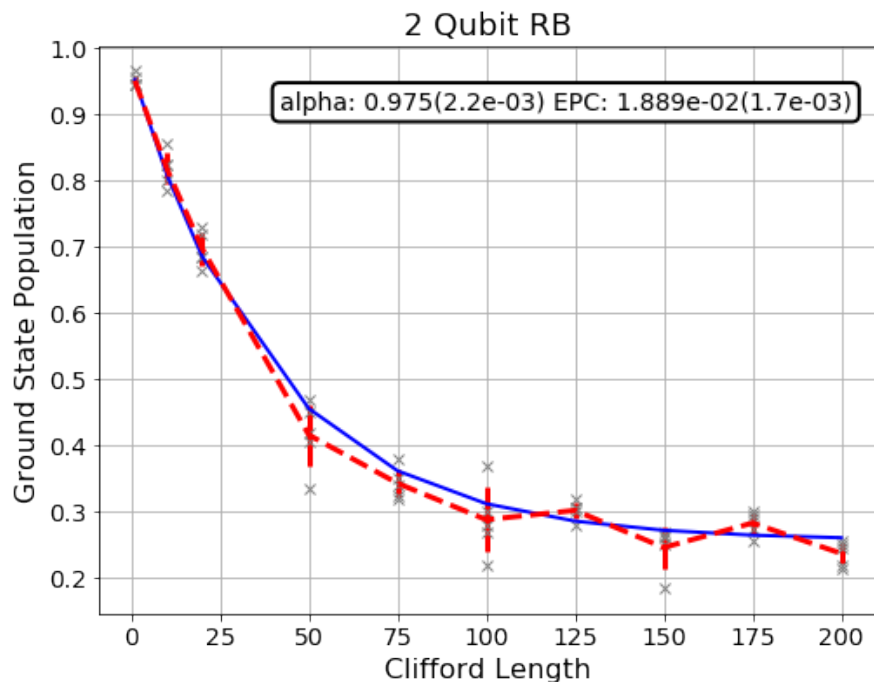
Fully characterizing all the quantum operations of our system is resource intensive. Instead we would like a general metric that tells us how our operations perform on average – for some definition of average.

Randomized benchmarking is widely used to measure an error rate of a set of quantum gates, by performing random circuits that would do nothing if the gates were perfect.



Exponential decay

In the limit of no finite-sampling error, the exponential decay rate of the observable survival probabilities, versus circuit length, yields a single error metric r .



For Clifford gates with arbitrary small errors described by process matrices, r corresponds to the mean, over all Cliffords, of the average gate infidelity between the imperfect gates and their ideal counterparts.

Fitting function: $Ar^m + B$

Error per Clifford: $EPC = \frac{2^n - 1}{2^n} (1 - r)$

Interpretation

For Clifford gates with arbitrary small errors described by process matrices, r corresponds to the mean, over all Cliffords, of the average gate infidelity between the imperfect gates and their ideal counterparts.



Protocol for single-qubit randomized benchmarking

(1) Choose a series of lengths for $L_1 < L_2 < \dots < L_m$ for your circuit

Do N_r times:

(2) For each L_i , generate a random sequence of $L_i - 3$ gates taken from $RB_{1Q} = \{I, RX(+/- 180^\circ), RX(+/- 90^\circ), RY(+/- 180^\circ), RY(+/- 90^\circ)\}$ and Calculate their completion A and its representation as a combination of 3 elements of RB_{1Q}

→ This produces a sequence of L_i elements of RB_{1Q} that compose to the identity operation

Do N_e times:

(3) Run the sequence of gates on your qubit and measure if you remain in state 0

→ You now have $N_e \times N_r \times m$ points of data

(4) Calculate $P_{L_i} = \{\text{probability of measuring 0 at sequence length } L_i\}$

(5) Fit $P_{L_i} = A + (B + CL_i)p^{L_i}$, where A , B , C and p are fit parameters

(6) Calculate $r = (1 - p)/2$ as the RB number