Development of an open-source calibration framework for superconducting qubits

Master degree in Physics

Candidate:

Elisa Stabilini

28326A

July 4th 2025

Università degli Studi di Milano - Department of Physics

Supervisor:

Prof. Dr. Stefano Carrazza

Co-supervisors:

Dr. Alessandro Candido

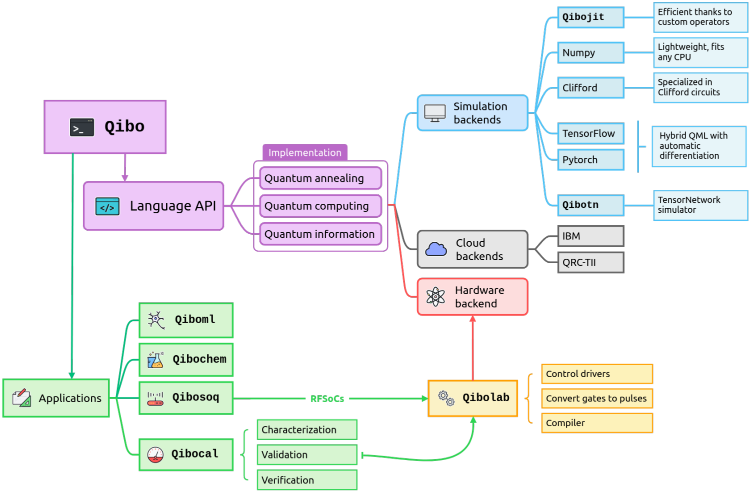
Dr. Andrea Pasquale

Dott. Edoardo Pedicillo

Technology Innovation Institute

Table of contents

Qibo framework



Superconducting qubits

Qubit: two level system

Superconducting qubits: use Josephson Junctions to build anharmonic oscillators

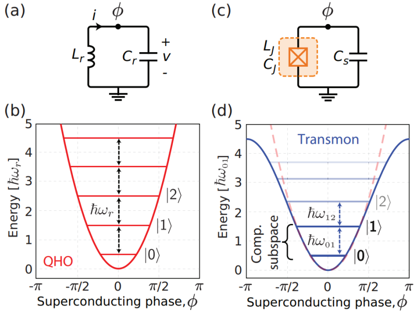


Figure 1: DOI: 10.1109/MAP.2022.3176593

Qubit control

Electromagnetic pulse applied to the qubit

$$V_d(t) = A\varepsilon(t)\sin\left(\omega_d t + \alpha\right),$$

Qubit - electric field Hamiltonian (RWA)

$$\hat{H} = -\frac{\hbar(\omega_q - \omega_d)}{2}\hat{\sigma}_z + \frac{\hbar\Omega}{2}\varepsilon(t)\left(\hat{\sigma}_x\cos\alpha + \hat{\sigma}_y\sin\alpha\right),$$

General qubit evolution under electromagnetic pulse:

$$R_{\hat{n}(\alpha)}(\theta) = e^{-\frac{i}{2}\hat{n}(\alpha)\cdot\vec{\sigma}\theta} = e^{-\frac{i}{2}(\hat{\sigma}_x\cos\alpha + \hat{\sigma}_y\sin\alpha)\theta},$$

where $\theta = \Omega\int_0^{+\infty}\varepsilon(t')dt'$

Qubit - resonator Hamiltonian:

$$\hat{H} = \hbar\omega_r \hat{a}\hat{a}^\dagger - \frac{\hbar\omega_{01}}{2}\hat{\sigma}_z + \hbar g(\hat{\sigma}^+\hat{a} + \hat{\sigma}^-\hat{a}^\dagger)$$

Dispersive regime ($g \ll \omega_q - \omega_r$):

$$\hat{H}_{disp} = \hbar(\omega_r - \chi\hat{\sigma}_z)\hat{a}^\dagger\hat{a} - \frac{\hbar}{2}(\omega_{01} + \chi)\hat{\sigma}_z$$

dispersive shift: $\chi = \frac{g^2}{\Delta}$,
$\Delta = \omega_q - \omega_r$

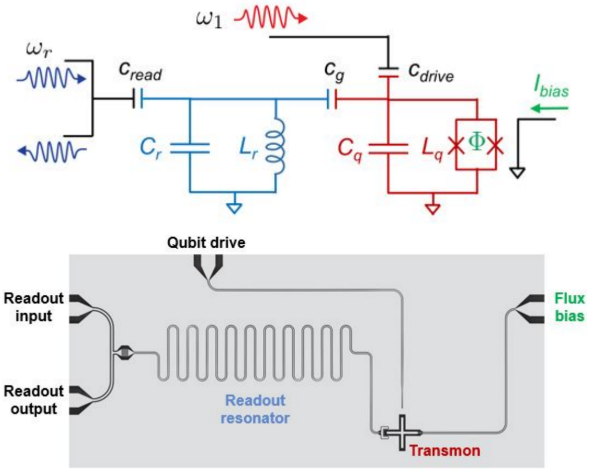
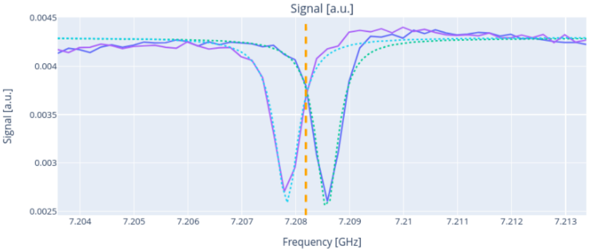
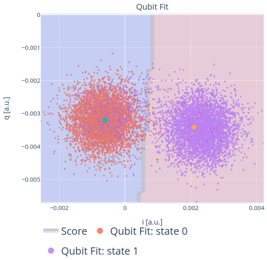


Figure 2: DOI: 10.1109/MAP.2022.3176593

Superconducting qubit calibration





Procedure:

1. Resonator characterization
2. Qubit characterization
3. Gate calibration
4. Gate set characterization

Metrics example:

- readout & assignment fidelity
- relaxation time T_1
- decoherence time $T_2 2$
- gate fidelity

Superconducting qubit calibration

Qubit	Readout Fidelity	Assignment Fidelity	T1 [μs]	T2 [μs]	Gate infidelity ($\cdot 10^{-3}$)
D1	0.876	0.938	26.4 ± 0.4	13.0 ± 2.0	27.0 ± 21.0
D2	0.945	0.973	14.9 ± 0.1	18.0 ± 10.0	20.5 ± 6.2
D3	0.905	0.952	23.6 ± 0.3	28.0 ± 12.0	7.0 ± 18
D4	0.929	0.964	20.6 ± 0.2	38.0 ± 5.2	4.4 ± 4.8
B2	0.902	0.951	17.5 ± 0.2	27.5 ± 5.2	18.0 ± 16

Procedure:

1. Resonator characterization
2. Qubit characterization
3. Gate calibration
4. Gate set characterization

Metrics example:

- readout & assignment fidelity
- relaxation time T_1
- decoherence time T_22
- gate fidelity

Average Clifford gate fidelity optimization

Randomized Benchmarking

Randomized benchmarking estimates average gate fidelity by applying random sequences of Clifford gates followed by an inverting gate.

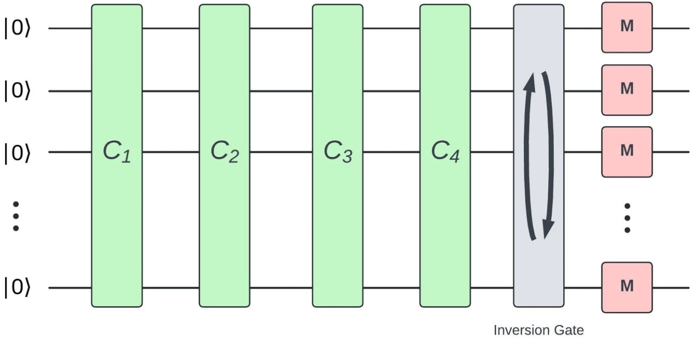
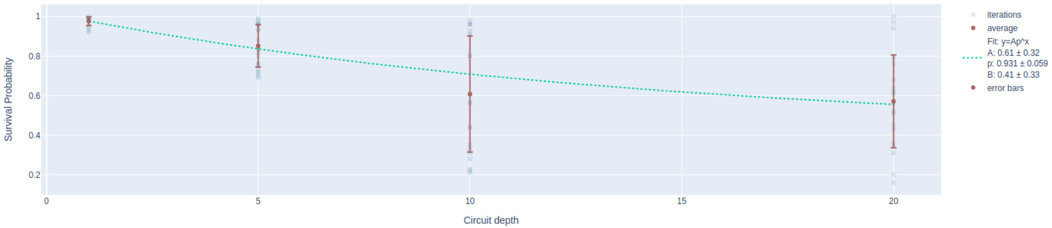


Figure 3: DOI: 10.1007/s10773-024-05811-8

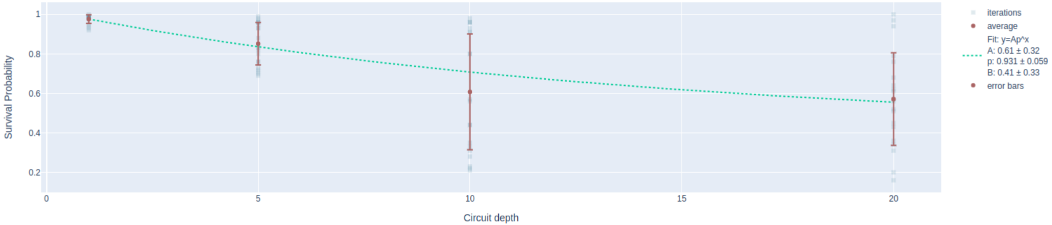
Randomized Benchmarking

Randomized benchmarking estimates average gate fidelity by applying random sequences of Clifford gates followed by an inverting gate.



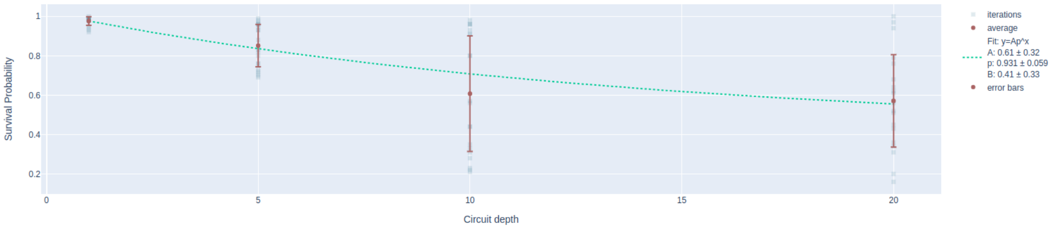
Randomized Benchmarking

Randomized benchmarking estimates average gate fidelity by applying random sequences of Clifford gates followed by an inverting gate.



Can we optimize the average gate fidelity to automate *RX* gate reacalibration?

RB optimization [Kelly et al. 2014]

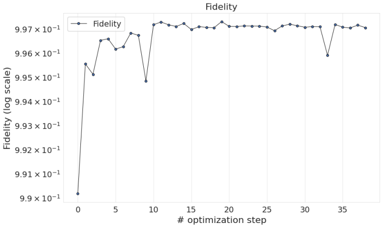


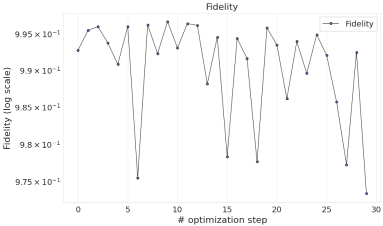
Can we optimize the average gate fidelity to automate RX gate reacalibration?

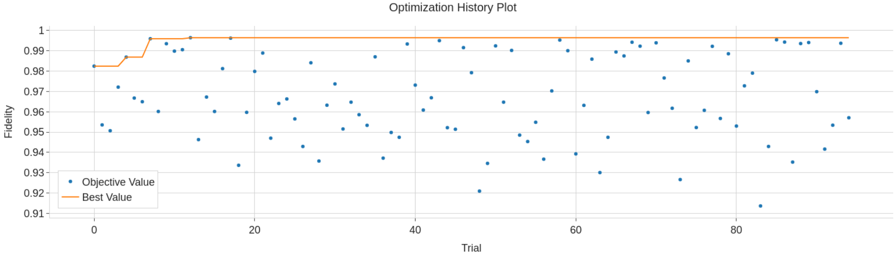
Test closed-loop optimization with modern optimization libraries

Optimization parameters: **amplitude**, **frequency** β **DRAG parameter**

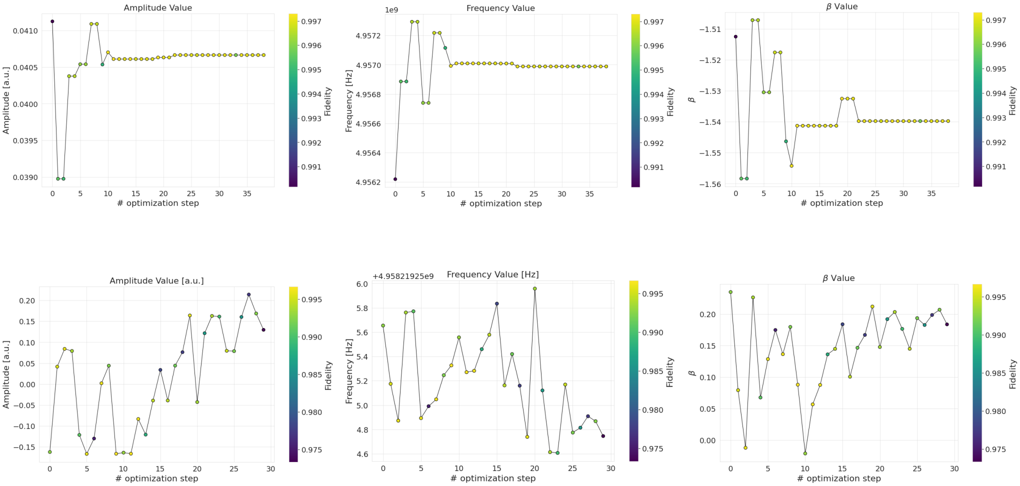
Average Clifford gate Fidelity







Parameters evolution



Library additions

Native RX90

Only native gates are directly implemented on hardware, other gates are transpiled into natives

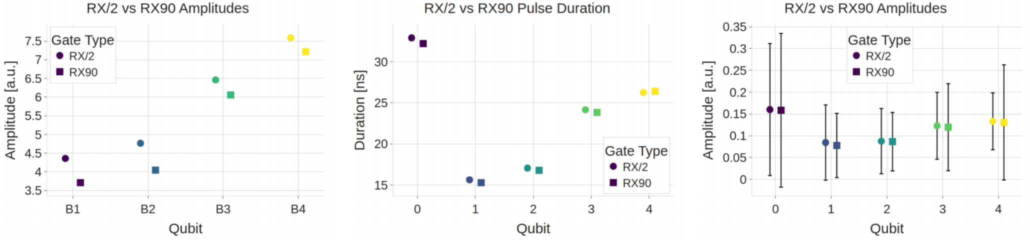
Qibolab single qubit natives: RX, $MZ + RX/2$

Native RX90

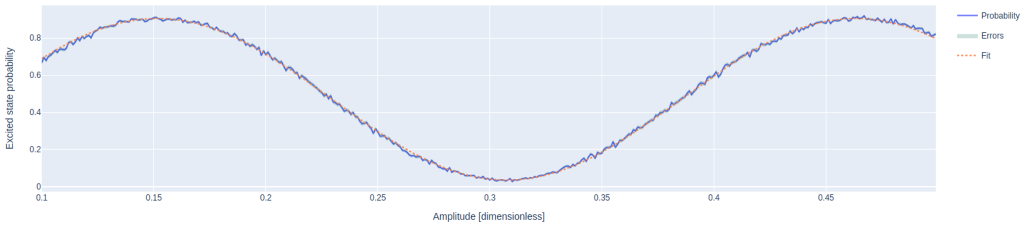
Only native gates are directly implemented on hardware, other gates are transpiled into natives

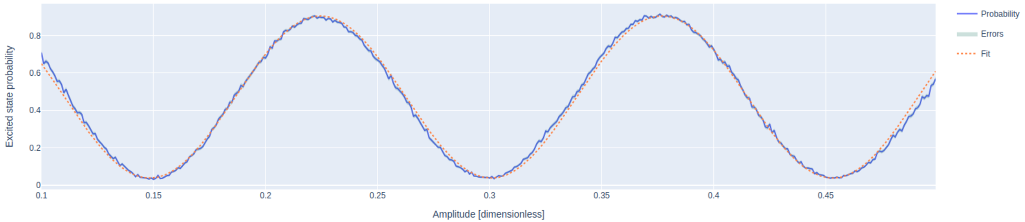
Qibolab single qubit natives: RX, MZ

Add RX90 native for more precise gates implementation



Rabi amplitude experiment





Flux pulse reconstruction [Rol et al. 2020]

Transmon flux dependence:

$$f_q(\Phi_q) \approx \left(\sqrt{8E_JE_C\left|\cos\left(\pi\frac{\Phi_q}{\Phi_0}\right)\right|}\right)$$

Detuning as function of the flux pulse:

$$\Delta f_R = \frac{1}{\Delta\tau}\int_{\tau}^{\tau+\Delta\tau} \Delta f_q(\Phi_{q,\tau+\Delta\tau}(t))$$

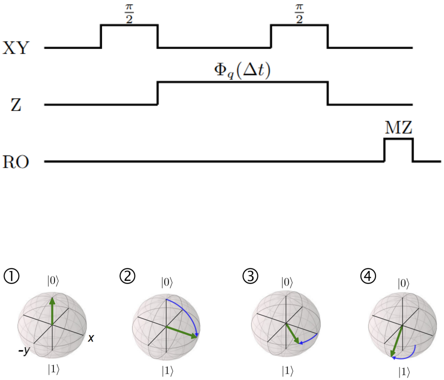
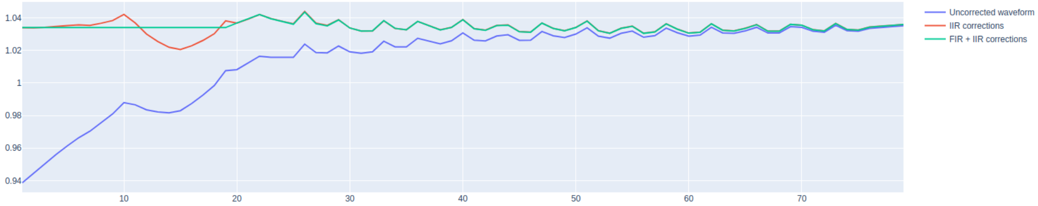


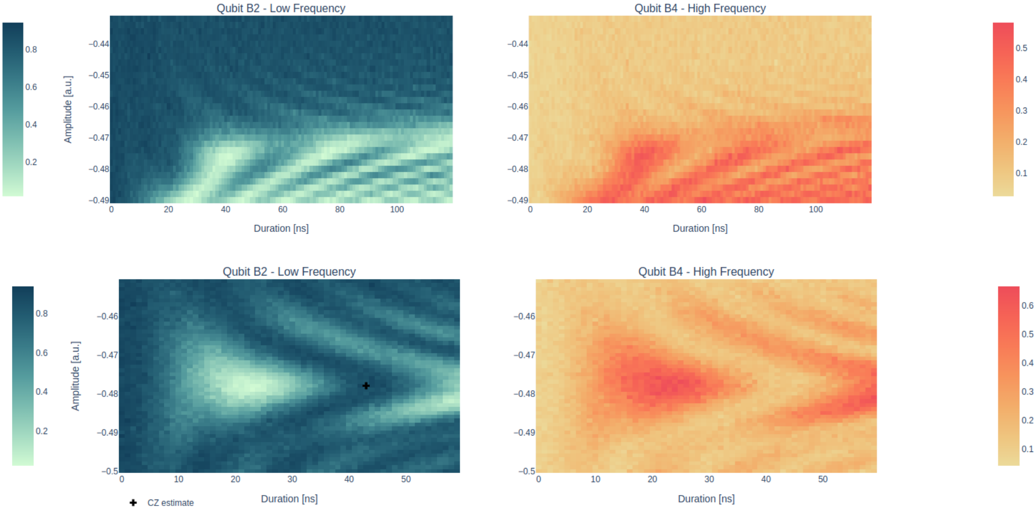
Figure 6: DOI: 10.1039/D2TC01258H

Filter determination



1. Determine exponential correction
2. Obtain IIR filters from exponential correction
3. Determine FIR
4. Apply pre-distortion

Impact on chevron plots



Cryoscope routine

Qibocal · v0.2.2

Search

relaxation_time: float

Wait time for the qubit to decohere back to the *gnd* state.

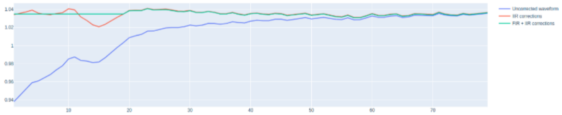
Example

A possible runcard to launch a Cryoscope experiment could be the following:

```
- id: cryoscope

  operation: cryoscope
  parameters:
    duration_max: 80
    duration_min: 1
    duration_step: 1
    flux_pulse_amplitude: 0.7
    relaxation_time: 50000
```

The expected output is the following:



> **Note**
>
> In the case where there are no filters the protocol will compute the FIR and the IIR filters. If the filters are already present the computation of the filters will be skipped and only the reconstructed waveform will be shown.

Requirements

- Single Shot Experiments

Conclusions & Outlooks

Conclusions & Outlooks

- Tested automated optimization routines to enhance single-qubit Clifford gates fidelities and highlighted main limitations for systematic application
- Possible future work related to parameters optimization of the RB protocol itself for this purpose
- Extended Qibolab and Qibocal to support native $R_X(\pi/2)$ gates with dedicated calibration routines
- Implemented and added the Cryoscope calibration experiment to Qibocal library
- Other possible experiments that can be added to Qibocal library include readout optimization, active qubit reset schemes, and leakage mitigation techniques

Thank you

References

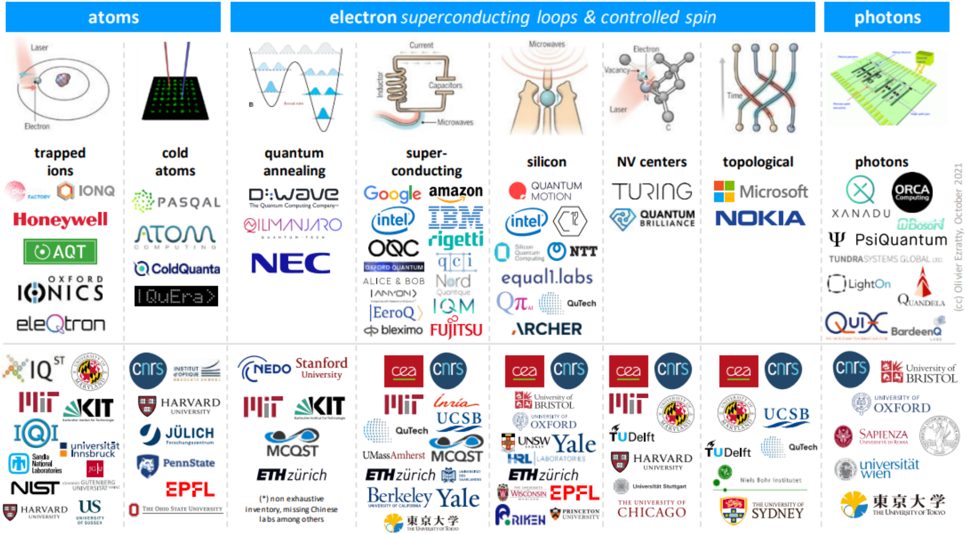
- Kelly, J. et al. (June 2014). **"Optimal Quantum Control Using Randomized Benchmarking".** en. In: *Physical Review Letters* 112.24, p. 240504. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.112.240504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.112.240504 (visited on 02/24/2025).
- Rol, M. A. et al. (Feb. 2020). **"Time-domain characterization and correction of on-chip distortion of control pulses in a quantum processor".** en. In: *Applied Physics Letters* 116.5. arXiv:1907.04818 [quant-ph], p. 054001. ISSN: 0003-6951, 1077-3118. DOI: 10.1063/1.5133894. URL: http://arxiv.org/abs/1907.04818 (visited on 02/24/2025).

Backup slides

What is for?

- **Simulation of quantum system:**
  "Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy"
- Optimization and modeling (chemistry, finance, traffic, weather...), eg. VQE, QAOA
- Quantum Algorithms
- Quantum Machine Learning

Qubit platforms



Standard Randomized Benchmarking protocol

RB protocol

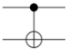
1. Initialize the system in the ground state
2. For each sequence length m draw a sequence of Clifford group elements
3. Calculate the inverse gate
4. Measure sequence and inverse gate
5. Repeat the process for multiple sequences of the same length while varying the length

RB features

- robust to SPAM errors
- faster than state tomography
- hardware-agnostic

Clifford gates

- Special subset of quantum gates that map Pauli operators to Pauli operators under conjugation
- Clifford gates group is generated by H, S, $(CNOT)$ gates
- Quantum circuits that consist of only Clifford gates can be efficiently simulated with a classical computer (Gottesman–Knill theorem)

Hadamard (H)		$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$
Phase (S, P)		$\begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$
Controlled Not (CNOT, CX)		$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix}$