Operational Quantum Tomography

SAND2019-5890C

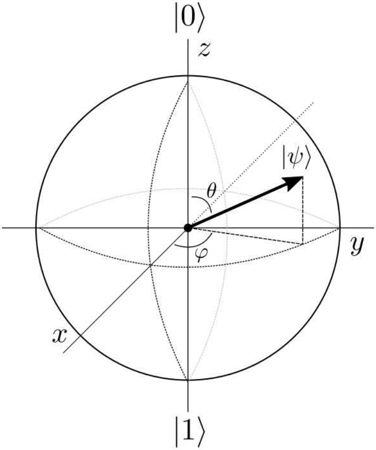
Olivia Di Matteo[1,2]

Joint work with:
John Gamble[2], Chris Granade[2], Kenneth Rudinger[3], Nathan Wiebe[2]

[1] TRIUMF, Vancouver, BC, Canada
[2] Microsoft Research, Redmond, WA, USA
[3] Quantum Performance Laboratory, Sandia National Laboratories, Albuquerque, NM, USA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Qubits



The two-level quantum systems that are used in *quantum computing*:

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle$$

where

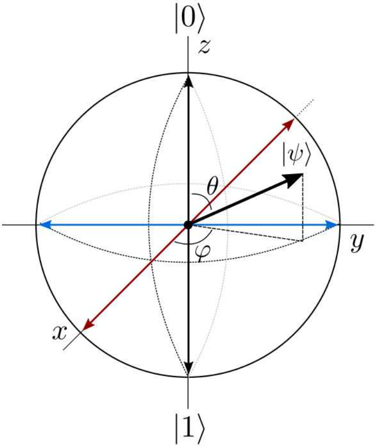
$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}, \quad |1\rangle = \begin{pmatrix}0\\1\end{pmatrix}.$$

Qubits can be represented in 3D as points in the *Bloch sphere*.

Given an unknown qubit state, how do we learn what it is?

Quantum state tomography

We can reconstruct a state by taking an informationally complete set of measurements.



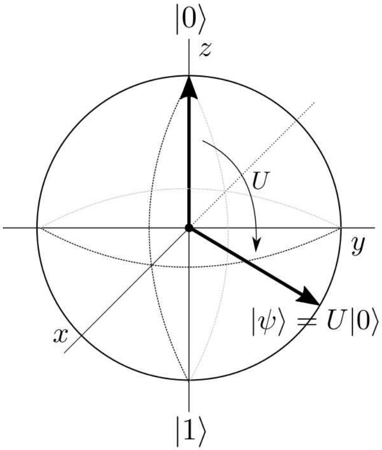
Measure:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Operating on qubits



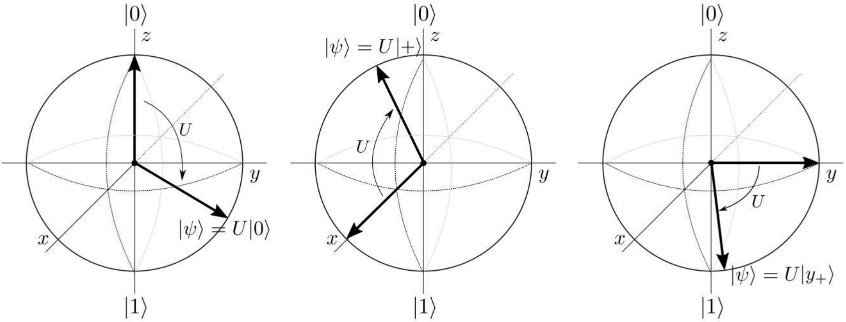
Qubit states are transformed using unitary operations, i.e. U such that

$$UU^\dagger = U^\dagger U = \mathbb{1}$$

Unitary operations rotate qubit states around the Bloch sphere.

How can we learn what an unknown unitary operation is doing?

Reconstruct an operation based on how it acts on known states.



QCVV: quantum characterization, verification, and validation

In the age of noisy quantum computers, it is important to characterize the behaviour of our quantum hardware.

Traditional quantum state and process tomography are done with very strong underlying assumptions:

- state tomography assumes measurements are perfect
- process tomography assumes initial states preparation *and* measurements are perfect

Are these reasonable assumptions?

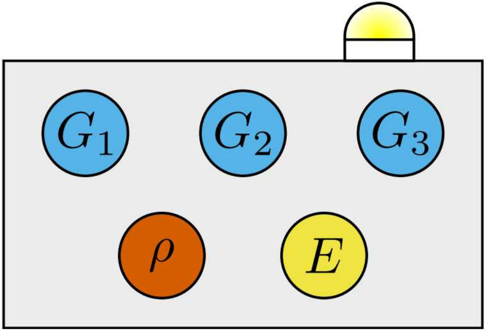
No - in real physical systems, **S**tate **P**reparation **A**nd **M**easurement (SPAM) are also noisy processes!

The results from our tomographic processes will not be *consistent* with each other, or with the true behaviour of the system.

So then... is there another means of learning about our system?

Gate set tomography (GST)*

Treat everything we can do to our quantum system equally.



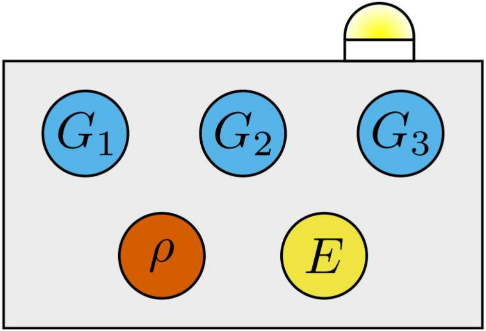
Learn about SPAM at the same time as the other processes we want to characterize.

Merkel, S. T., et al. (2013). *Self-consistent quantum process tomography.* Physical Review A, 87(6).

Blume-Kohout, R., Gamble, J. K., Nielsen, E., Mizrahi, J., Sterk, J. D., & Maunz, P. (2013). *Robust, self-consistent, closed-form tomography of quantum logic gates on a trapped ion qubit.* `http://arxiv.org/abs/1310.4492`

Gate set tomography (GST)

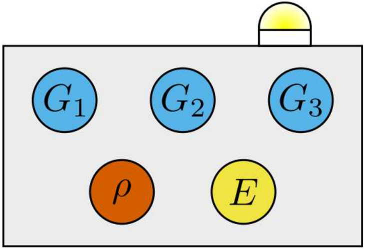
We represent state preparation, unitaries (gates), and measurement as ‘buttons’ that we can push to operate on our quantum system.



A light on the box either turns on, or stays off, to indicate the outcome of the measurement.

Gate set tomography (GST)

Mathematically, we will represent every button as a *superoperator* - our initial task will be to learn their contents.



$$|\rho\rangle\rangle = \begin{pmatrix} * & * & * & * \end{pmatrix}^T$$

$$|E\rangle\rangle = \begin{pmatrix} * & * & * & * \end{pmatrix}^T$$

$$G_1 = \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix}$$

$$G_2 = \ldots$$

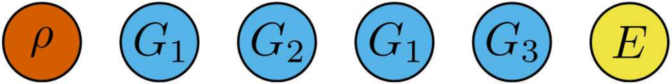
Important assumption: a button has the same action (i.e. same superoperator) every time it is pressed.

How do we learn our superoperators?

GST experiments

By pushing a bunch of buttons and chosen in a clever way.

GST experiments take the following form:

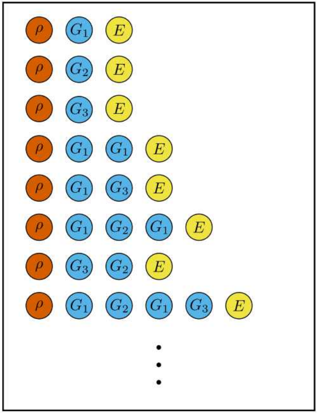


Experiments are performed multiple times - we record the frequency with which the light turned on.

Linear-inversion GST

We can reconstruct the superoperators by using the outcome frequencies from a variety of experiments.

But wait... we know *nothing* about the system; we have only the assumption that each button performs the same action whenever it's pressed.

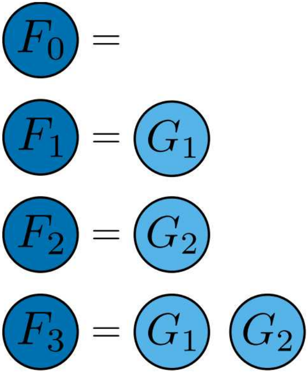


Linear-inversion GST

We perform GST with respect to a set of *fiducial experiments*, or *fiducial sequences*.

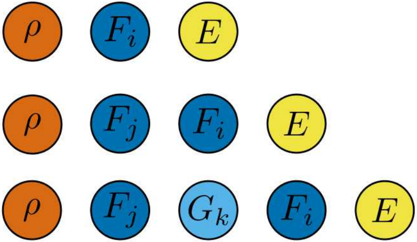
These are short sequences of only one or two button presses that gives us a point of reference.

The set of fiducial sequences must be informationally complete (more on this in a minute!)



We perform experiments using the fiducial sequences to construct a set of objects:

$$\tilde{E}, \quad \tilde{F}, \quad \{\tilde{G}^{(k)}\}$$



$$\tilde{E}_i = \langle\langle E|F_i|\rho\rangle\rangle$$

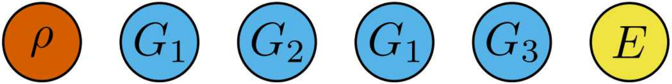
$$\tilde{F}_{ij} = \langle\langle E|F_i F_j|\rho\rangle\rangle$$

$$\tilde{G}_{ij}^{(k)} = \langle\langle E|F_i G_k F_j|\rho\rangle\rangle$$

Linear-inversion GST

The matrix elements $\tilde{E}_i, \tilde{F}_{ij}, \tilde{G}_{ij}^{(k)}$ represent *sequence probabilities* - they can be calculated analytically from the superoperators using Born's rule.

For example, consider the experiment:

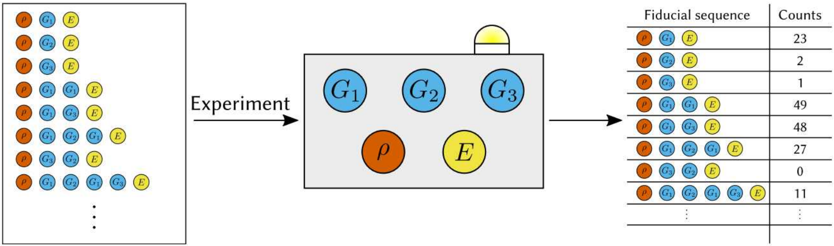


The probability that the light turns on is

$$\langle\langle E | G_3 G_1 G_2 G_1 | \rho \rangle\rangle = \mathrm{Tr}\left(| \rho \rangle\rangle \langle\langle E | G_3 G_1 G_2 G_1 \right)$$

Linear-inversion GST

Repeatedly run the experiments for the fiducial sequences, and look at the frequency with which the light turned on.



Use the obtained frequencies to populate $\tilde{E}$, $\tilde{F}$, and $\{G^{(k)}\}$.

Linear-inversion GST

In principle, we can perform *linear inversion* to obtain our $|\rho\rangle\rangle,\ |E\rangle\rangle,\ \{G_k\}$ from the $\tilde{E},\ \tilde{F},\ \{\tilde{G}^{(k)}\}$:

$$\begin{aligned}
\tilde{E}_i &= \langle\langle E|F_i|\rho\rangle\rangle & & \tilde{E}^T = \langle\langle E|B \\
\tilde{F}_{ij} &= \langle\langle E|F_iF_j|\rho\rangle\rangle & \Rightarrow\quad & \tilde{F}^{-1}\tilde{E} = B^{-1}|\rho\rangle\rangle \\
\tilde{G}^{(k)}_{ij} &= \langle\langle E|F_iG_kF_j|\rho\rangle\rangle & & \tilde{F}^{-1}\tilde{G}^{(k)} = B^{-1}G_kB
\end{aligned}$$

We can learn our superoperators up to some additional linear transformation B. For this to work out analytically, $B_{ij} = (F_j|\rho\rangle\rangle)_i$.

This is where we can define 'informationally complete' - we need to choose the fiducials so that $\tilde{F}$ is invertible (and for a single qubit, there must be four of them)!

Linear-inversion gateset tomography

But there's a problem... Recall the Born rule and the sequence probability. We want to learn $\rho, \; E, \; \{G_k\}$, but after linear inversion we have expressions for them up to some matrix B.

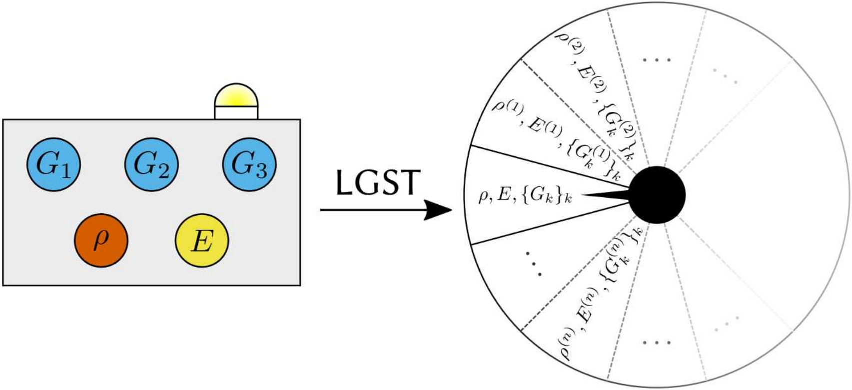
B is *not accessible* experimentally!

$$\mathrm{Tr}\left(|\rho\rangle\rangle\langle\langle E| G_{s_k} \cdots G_{s_1}\right) = \mathrm{Tr}\left(B^{-1}|\rho\rangle\rangle\langle\langle E| B B^{-1} G_{s_k} B \cdots B^{-1} G_{s_1} B\right)$$

The sequence probability doesn't change with the inclusion of B. B is a *unknown gauge transformation.*

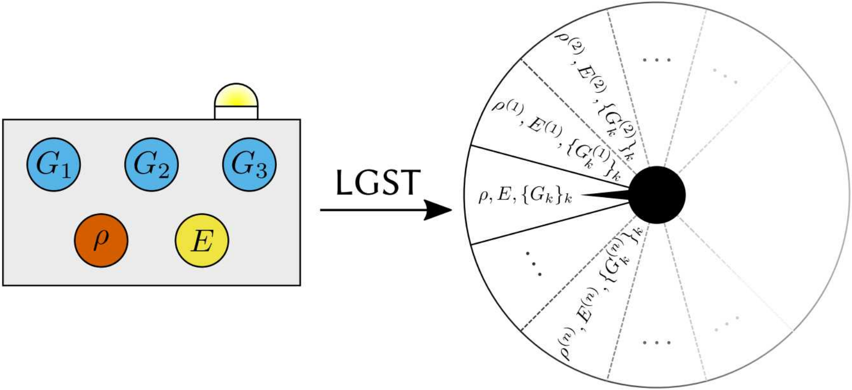
Gauge freedom in gateset tomography

If we can't access B, we don't know the 'true' superoperators, we only learn one set in their *gauge orbit*. Superoperators are a gauge-dependent quantity.



Gauge freedom in gateset tomography

If we can't access B, we don't know the 'true' superoperators, we only learn one set in their *gauge orbit*. Superoperators are a gauge-dependent quantity.



...what can we do about this?

Gauge freedom in gateset tomography

Option 1: Gauge-fixing.
Run a computational procedure to find a B that makes your superoperators close to what you think they should be.

Gauge freedom in gateset tomography

Option 1: Gauge-fixing.
Run a computational procedure to find a B that makes your superoperators close to what you think they should be.

Issues:

- May be computationally costly
- Requires assumptions about the action of the buttons

Option 1: Gauge-fixing.
Run a computational procedure to find a B that makes your superoperators close to what you think they should be.

Issues:

- May be computationally costly
- Requires assumptions about the action of the buttons

Option 2: Work with gauge-*independent* quantities instead.

An operational representation

Recall the sequence probabilities we obtained from experiment:

$$
\begin{aligned}
\tilde{E}_i &= \langle\langle E|F_i|\rho\rangle\rangle \\
\tilde{F}_{ij} &= \langle\langle E|F_i F_j|\rho\rangle\rangle \\
\tilde{G}_{ij}^{(k)} &= \langle\langle E|F_i G_k F_j|\rho\rangle\rangle
\end{aligned}
$$

These quantities, and consequently $\tilde{E}$, $\tilde{F}$, and $\tilde{G}^{(k)}$, are gauge-*independent*.

An operational representation

Furthermore, we can use them in the general expression to compute arbitrary sequence probabilities...

$$
\begin{aligned}
\Pr(\text{light}) &= \operatorname{Tr}\left(|\rho\rangle\rangle\langle\langle E| G_{s_k} \cdots G_{s_1}\right) \\
&= \operatorname{Tr}\left(B^{-1}|\rho\rangle\rangle\langle\langle E| B B^{-1} G_{s_k} B \cdots B^{-1} G_{s_1} B\right) \\
&= \operatorname{Tr}\left(B^{-1}|\rho\rangle\rangle\langle\langle E| B B^{-1} G_{s_k} B \cdots B^{-1} G_{s_1} B\right) \\
&= \operatorname{Tr}\left(\tilde{F}^{-1}\tilde{E} \cdot \tilde{E}^T \cdot \tilde{F}^{-1}\tilde{G}^{(s_k)} \cdot \tilde{F}^{-1}\tilde{G}^{(s_{k-1})} \cdots \tilde{F}^{-1}\tilde{G}^{(s_1)}\right)
\end{aligned}
$$

If we can learn $\tilde{E}$, $\tilde{F}$, and $\tilde{G}^{(k)}$, we can predict the outcome of any future experiment! We call them the **operational representation**.

Bayesian inference for the operational representation

But how do we actually go about learning it from the data?

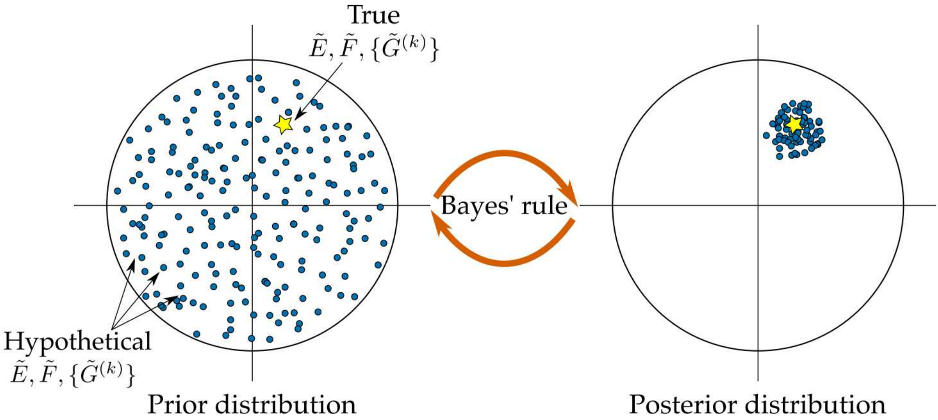
We will use Bayesian inference. Bayes rule tells us that:

$$\Pr(\text{model} \mid \text{data}) \propto \Pr(\text{data} \mid \text{model}) \cdot \Pr(\text{model})$$

We will create a *prior distribution* of a large number of *hypothetical* versions of $\tilde{E}$, $\tilde{F}$, and $\tilde{G}^{(k)}$, and then perform Bayesian updates using data to obtain a *posterior distribution*.

We call this operational quantum tomography (OQT).

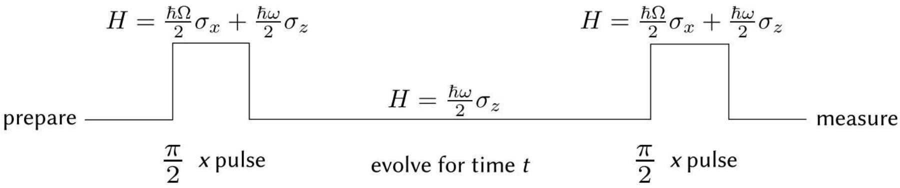
Bayesian inference for the operational representation



Example: Ramsey interferometry

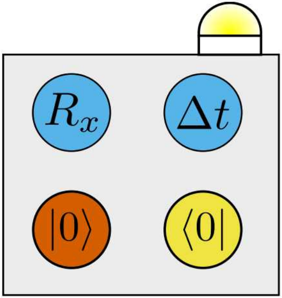
The Rabi oscillation frequency (ω) tells us the likelihood of a qubit being in either $|0\rangle$ or $|1\rangle$ in the presence of a driving field.

Single-qubit operations are often implemented by applying EM pulses to induce rotations around the Bloch sphere. Knowing the Rabi frequency helps us select the pulse duration that performs a desired operation. We can learn it using *Ramsey interferometry*:



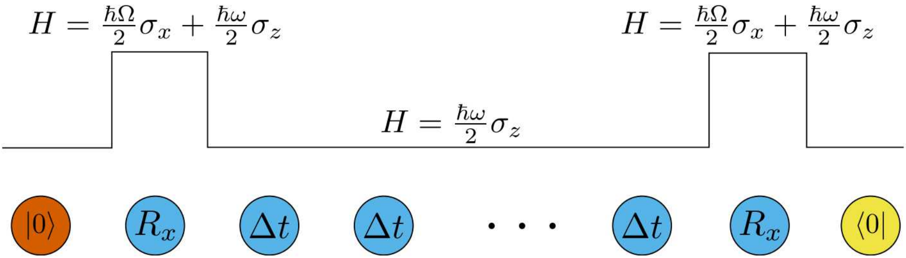
We can express Ramsey interferometry in the OQT formalism to help us learn the oscillation frequency.

We'll have to 'discretize' time to represent it as a button press.



OQT for Ramsey interferometry

The full experiment then looks something like this:



OQT for Ramsey interferometry

The OQT process for Ramsey interferometry consists of the following steps:

1. Choose a **prior distribution** for what we think the operational representations should look like.

   It's not obvious what properties an arbitrary operational representation should have, except it should contain all positive numbers.

   Instead, we choose a prior over the *superoperators*, e.g.

   $$R_x(\frac{\pi}{2}) \to R_x(\frac{\pi}{2} + \epsilon), \quad \epsilon \in \mathcal{N}(0, \sigma^2)$$

   and use these to later convert to the gauge-independent form.

OQT for Ramsey interferometry

2. Choose a set of fiducial sequences.

   This can be done using trial and error to see what 'works', i.e. makes $\tilde{F}$ invertible.

   We chose:

   $$F_0 = \qquad\qquad F_2 = R_x \; R_x$$

   $$F_1 = R_x \qquad F_3 = R_x \; \Delta t \; R_x$$

   An 'empty' fiducial indicates an experiment where we perform only SPAM.

OQT for Ramsey interferometry

3. Initialize a particle cloud with many hypothetical operational representations.

   Sample superoperators from their priors. In the following example, we initialize a cloud of 10000 particles under the following assumptions:

   - State preparation is perfect: $|0\rangle$[1]
   - Measurement in computational basis is perfect
   - $R_x(\frac{\pi}{2})$ pulled from $R_x(\frac{\pi}{2}+\epsilon)$, $\epsilon \in \mathcal{N}(0, 10^{-3})$
   - Δt pulled from $R_z(\omega \cdot dt)$, $\omega \in [0, 1]$, $dt = 1$

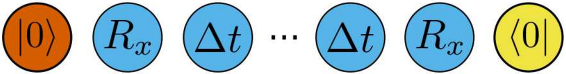
   Combine and use the fiducial sequences to compute $\tilde{E}$, $\tilde{F}$, and $\tilde{G}^{(k)}$ for each individual sample.

[1]Yes, I previously explained that these assumptions are bad, but this is for simplicity - I will show another example where this is not the case!

OQT for Ramsey interferometry

4. Perform Bayesian inference

   Using either true experimental data, or simulated data, perform a series of experiments of the following form:



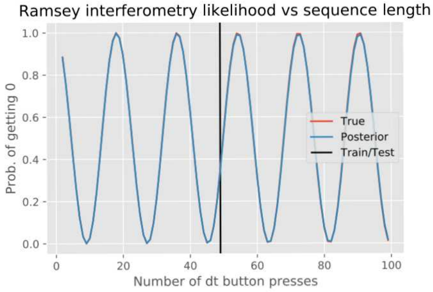
   We performed experiments starting at 2 Δt presses, up to 50.

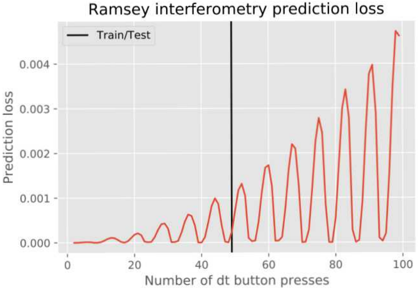
   For each n, perform the experiments, and update the particle cloud of hypothetical operational representations according to Bayes' rule[2].

[2] We used Sequential Monte Carlo techniques to do this.

5. Assess the quality of our reconstruction.

   Use the posterior distribution after training up to $n = 50$ button presses, and estimate sequence probabilities up to $n = 100$.



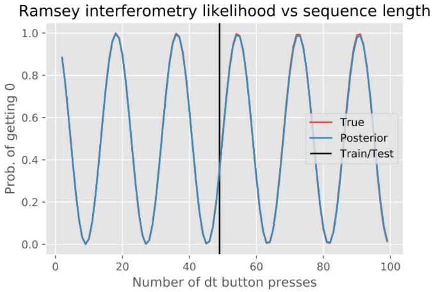


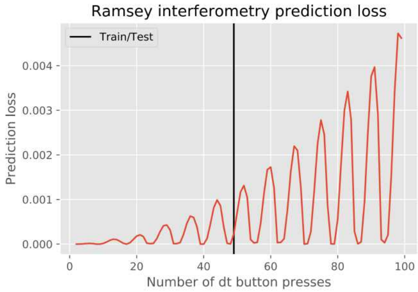
OQT for Ramsey interferometry

5. Assess the quality of our reconstruction.

We use a prediction loss: for true probability p_s and estimated probability $\hat{p}_s$, the loss is given by

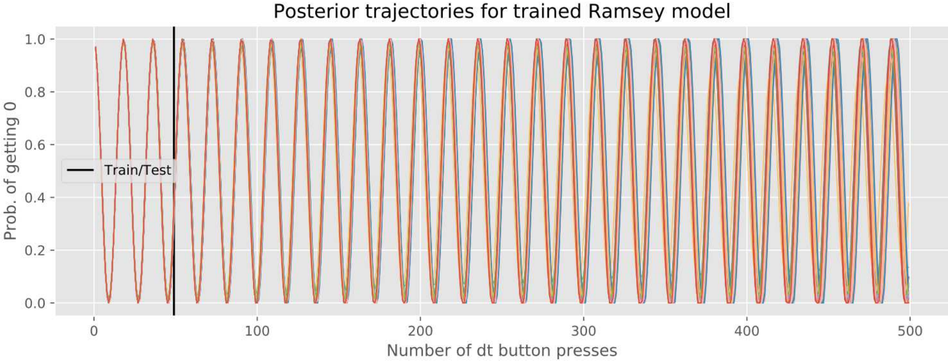
$$\text{Loss}(p_s, \hat{p}_s) = (\hat{p}_s - p_s)^2$$





OQT for Ramsey interferometry

The curves shown above are for a mean operational representation computed over the whole posterior. In fact, each particle in the posterior gives a slightly different trajectory.



Why is this interesting?

OQT for Ramsey interferometry

Why is this interesting?

Ramsey interferometry is *not something that can be addressed* using standard GST techniques.

OQT for Ramsey interferometry

Why is this interesting?

Ramsey interferometry is *not something that can be addressed* using standard GST techniques.

We find in general that OQT is applicable to a broad array of characterization tasks.

OQT for Ramsey interferometry

Why is this interesting?

Ramsey interferometry is *not something that can be addressed* using standard GST techniques.

We find in general that OQT is applicable to a broad array of characterization tasks.

So far we have also successfully performed:

- Quantum state tomography
- Quantum process tomography (with simulated and real data)
- Randomized benchmarking

OQT for experimental trapped-ion qubit data

Long-sequence gateset tomography: take linear-inversion GST as a starting point, perform additional experiments and update estimates using maximum likelihood techniques.

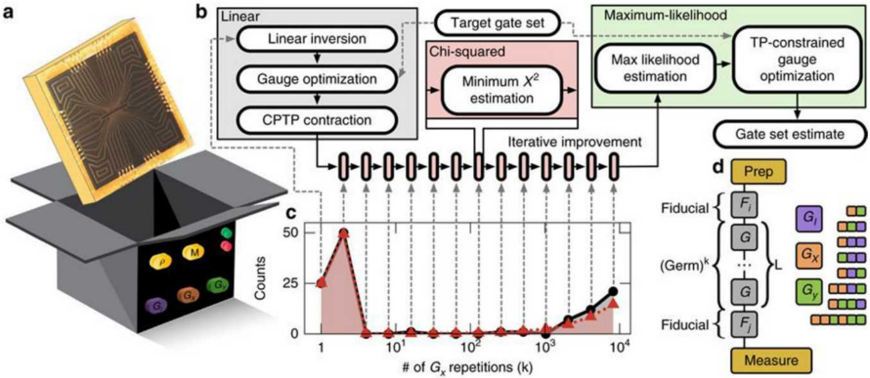
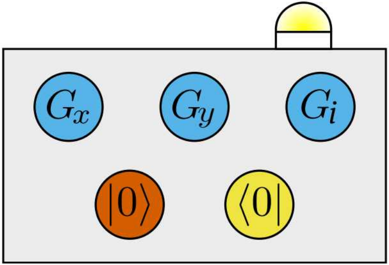


Image: Blume-Kohout, R., Gamble, J. K., Nielsen, E., Rudinger, K., Mizrahi, J., Fortier, K., & Maunz, P. (2017). *Demonstration of qubit operations below a rigorous fault tolerance threshold with gate set tomography.* Nature Communications, 8, 14485.

OQT for experimental trapped-ion qubit data

We perform OQT using the same experimental data and same sequence of experiments as performed in long-sequence GST.

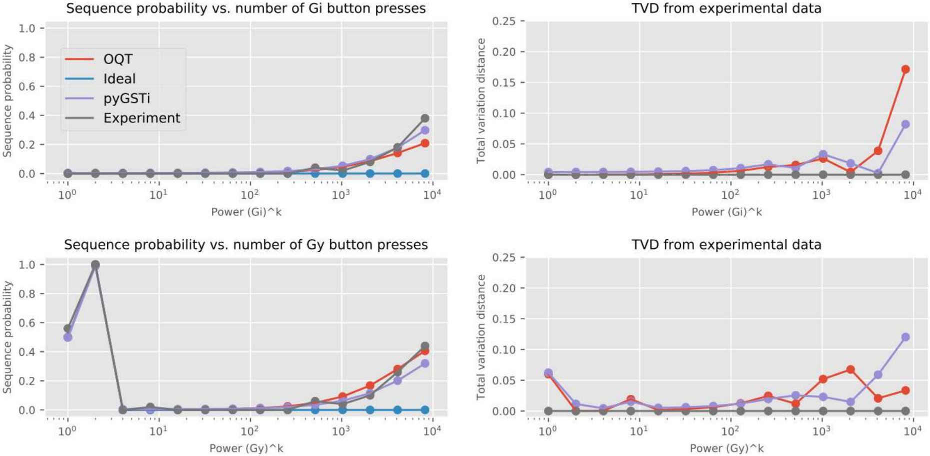


Experimental operations will be noisy versions of:

- G_i, the identity gate
- G_x, a $\frac{\pi}{2}$ rotation about x
- G_y, a $\frac{\pi}{2}$ rotation about y
- preparing the state $|0\rangle$
- measurement in the computational basis

OQT for experimental trapped-ion qubit data

We find that our results are competitive with existing techniques!



Conclusions and future work

Operational tomography allows us to characterize and learn about a wide variety of quantum systems.

Learning the operational representation allows us to predict the outcome of future experiments in a way that is independent of the gauge-related difficulties suffered by other procedures.

Next steps for OQT:

- Scaling up to multi-qubit systems
- Multi-state / multi-measurement cases

Thank you for your attention!

Thanks also to:

- My co-authors John, Chris, Kenny, and Nathan
- QuArC group at Microsoft Research, for hosting me as an intern and frequent visitor