Error mitigation

Adrián Pérez Salinas

[1]Lorentz Institute - Leiden University
[2]$\langle aQa \rangle^L$: Applied Quantum Algorithms

Spring school in near-term quantum computing
— Benasque 2024 —

Contents

Section 1

Introduction

Introduction

- All computations (classical or quantum) are susceptible to suffer errors due to hardware faults
- To avoid computational errors, correction schemes exist

An easy example

- Consider a classical bit (0/1)
- Its physical support may suffer an spontaneous flip (for any reason) with probability p
- The probability of failure is p

Redundancy encoding

Encode the logical bit into many bits: $1 \to 111\ldots1111$, of length n, and we apply a majority rule

What is the probability of logical fault?[a]

$$\mathrm{Prob}_{\mathrm{fault}}(n, p) \leq \exp\left(-2n\left(\frac{1}{2} - p\right)^2\right)$$

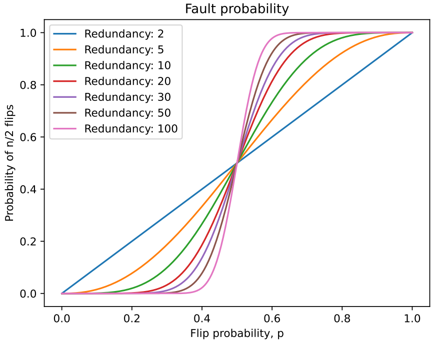
[a] Available through binomial distribution and Hoeffding's inequality

Introduction

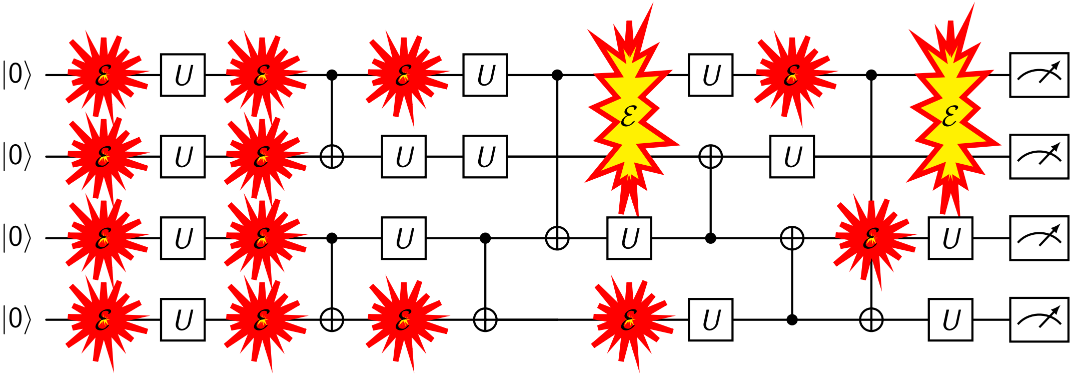
- All computations (classical or quantum) are susceptible to suffer errors due to hardware faults
- To avoid computational errors, correction schemes exist

An easy example

- Consider a classical bit (0/1)
- Its physical support may suffer an spontaneous flip (for any reason) with probability p
- What is the probability of flipping the entire memory?



Effect of noise in a quantum computer



Noise affects the capabilities of quantum computing

- Too much noise makes quantum computing classically simulable
- Even in a fault-tolerant regime, the overhead to correct errors might kill any quantum advantage
- Even in a fault-tolerant regime, quantum computers cannot be free of sampling (shot) noise

Noise is ubiquitous in all devices

- First quantum *supremacy* experiment achieved a fidelity of ~ 0.002
- All quantum computers suffer from decoherence, non-perfect fidelity, cross-talk...
- We cannot expect any device (even with advanced technology) to provide 100% accurate quantum algorithms

Noise handling in quantum computer

Quantum Error Mitigation

- Decrease error of a measured observable at the end of computation
- Low qubit overhead
- Low circuit runtime
- Exponential sampling overhead (scaling with error)
- Keep it low for performance
- Mid-circuit measurements are not required (or infrequent)

Available now!

Current experiments are already running these techniques

Quantum Error Correction

- Detect and correct errors in the state on the fly
- High qubit overhead
- Runtime depends on the circuit
- Constant sampling overhead
- Error-rate must be below threshold
- Mid-circuit measurements are essential

Available in the future

Many codes exist, but their large-scale implementation needs to wait until low error levels, below threshold

This lecture

- What kind of noises exist
- What techniques can we use for mitigating noise
- What are the overheads imposed by error mitigation
- What is the final goal, a. k. a. error correction

Section 2

Kinds of noise

Noise in quantum computing

Noise models are CPTP[a] maps

$$\mathcal{E}(\rho) \to \left(1 - \sum_i p_i\right) \rho + \sum_i p_i A_i \rho A_i^\dagger$$

under the condition $\sum_i p_i A_i A_i^\dagger \leq I \sum_i p_i$.
The operators A_i are usually called Kraus operators

[a] Completely positive trace preserving

Some commonly used examples

Depolarizing noise

In d dimensions: $\mathcal{E}(\rho) = (1-p)\rho + \frac{p}{d} I$

Pauli noise

Valid or single-qubit systems, $A_i = \sigma_i$
If $p_1 = p_2 = p_3$, the Pauli channel is depolarizing

Decoherence and dephasing

$K_z = Z$, $K_0 = |0\rangle \langle 0|$
This model is physically motivated through thermal relaxation, and $p_i \approx 1 - e^{-t/T_i}$.

General considerations of noise

- Noise generically transforms pure states into mixed states
- Noise induces undesirable features that can be detected (e.g. symmetry breaking)
- Modeling noise is useful, but not all noise is easily captured (non-markovian noise, cross-talk, ...)
- Noise can depend on many parameters
- Having perfect knowledge of single pieces does not always guarantee knowledge of the entire system , i. e. there are emergent properties

Section 3

Error mitigation

Common framework

Terminology

- *Primary circuit*: the circuit of interest, assuming no error
- *Perfect output state*: output of the primary circuit, ρ_0
- *Noisy state*: output of the primary circuit, affected by noise, ρ
- *Observable of interest*: measurement done on the noisy state $O \to \mathrm{Tr}(O\rho)$
- *Shots*: Number of repetitions of the circuit, N
- *Shot noise*: noise inherent to probabilistic nature of quantum computing
- *Sampling overhead*: Excess in N required to perform error mitigation, C_{em}

Goal

Obtain the most accurate values for

$$\mathrm{Tr}(O\rho) \approx \mathrm{Tr}(O\rho_0)$$

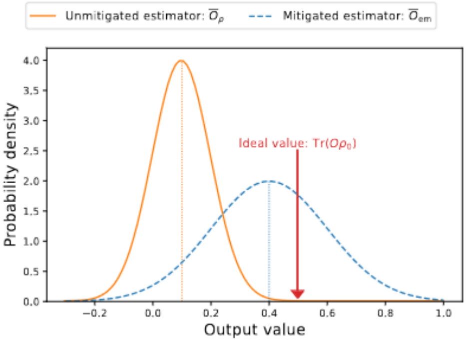
Common framework

We aim to find an estimator $\hat{O}$ reducing the error[a]

$$E(\hat{O}) = \mathbb{E}\left[\left(\hat{O} - \mathrm{Tr}(O\rho)\right)^2\right],$$

which we can split in two pieces by identifying $\mathbb{E}[\hat{O}] = \mathrm{Tr}(O\rho)$

$$E(\hat{O}) = \underbrace{(\mathrm{Tr}(O\rho) - \mathrm{Tr}(O\rho_0))^2}_{\text{bias}} + \underbrace{\frac{\mathrm{Tr}\left(O^2\rho\right) - \mathrm{Tr}(O\rho)^2}{N}}_{\text{Variance}}$$



[a] $\mathbb{E}$ denotes expectation values

Common framework

Goal: reduce the bias

- Achieved through constructing more complex estimators $\hat{O}_{em}$
- These estimators have larger variances
- We need to increase the number of shots to compensate

$$C_{rm} = \frac{N(\hat{O}_{em})}{N(\hat{O}_\rho)} = \frac{\mathrm{Var}[\hat{O}_{em}]}{\mathrm{Var}[\hat{O}_\rho]}$$

Probability of faults

$$P_0 = \prod_{f=1}^{F} (1 - p_f) \approx (1 - p)^F.$$

Alternatively, the average number of faults is

$$\lambda = \sum_f p_f.$$

For small p_f and many possible faults, one can use Poisson's distribution to obtain

$$P_0 \approx e^{-\lambda},$$

which hints for exponential overheads.

Zero-noise extrapolation

A mathematical trick: Richardson extrapolation Let $A_0(x) \approx A^*$, such

$$A^* = A_0(h) + a_0 h^{k_0} + a_1 h^{k_1} + a_2 h^{k_2} + a_3 h^{k_3} + \dots$$

in decreasing order $h^{k_i} \geq h^{k_{i+1}}$. h is the step.
We aim to decrease the error of approximationby comparing two step sizes $h, h/t$ for a positive constant t. Then

$$A^* = A_1(h) + a_1 h^{k_1} + a_2 h^{k_2} + a_3 h^{k_3} + \dots$$

and we can obtain a recursive formula

$$A_i(h) = \frac{t^{k_i} A_{i-1}(h/t) - A_{i-1}(h)}{t^{k_{i-1}} - 1}$$

Zero-noise extrapolation

We apply Richardson trick by artificially increasing the fault rate λ, and fitting

$$\mathrm{Tr}(O\rho_\lambda) = f(\lambda, \vec{\theta}),$$

for tunable $\vec{\theta}$. We aim for the zero-noise value $f(0, \vec{\theta})$.
Assuming small λ we can find a good polynomial approximation,

$$f(\lambda, \vec{\theta}) = \sum_{m=0}^{M-1} \theta_m \frac{\lambda^m}{m!}$$

A closed formula from sets $\{\lambda_m, \mathrm{Tr}(O\rho_{\lambda_m})\}$

$$\mathbb{E}[\hat{O}] = \sum_{m=1}^{M} \mathrm{Tr}(O\rho_{\lambda_m}) \prod_{k\neq m} \frac{\lambda_k}{\lambda_k - \lambda_m}$$

Choose your λ_m wisely! Bad choices may blow up the overhead

$$C_{em} = \left(\sum_{m=1}^{M} \left| \prod_{k\neq m} \frac{\lambda_k}{\lambda_k - \lambda_m} \right| \right)^2$$

Zero-noise extrapolation

- Theoretical guarantees for small λ, practical performance for large λ
- Divergence at $\lambda \to \infty$, combine with exponential extrapolation (use the function $e^{\lambda}\,\mathrm{Tr}(O\rho_\lambda)$)
- Difficult problem: how to increase λ without losing the characterization of the noise models?
  - Adding more gates with no noiseless effect?
  - Engineeringly tune the error parameters?
- Can we effectively learn the noise models?

Probabilistic error cancellation

- Any noisy circuit is a modified primary quantum circuit, and we can expect to estimate the noiseless observable as a linear combination of noisy circuits.
- Then, by Monte Carlo sampling, we can recover a more accurate estimator, even with zero bias

Let $\mathcal{U}$ be an ideal channel and $\{\mathcal{B}_n\}$ a basis of noisy operations, then

$$\mathcal{U} = \sum_n \alpha_n \mathcal{B}_n$$

and then

$$\mathrm{Tr}(O\rho_0) = \mathrm{Tr}(O\mathcal{U}(\rho_{\mathrm{in}})) = \sum_n \alpha_n \mathrm{Tr}(\mathcal{B}_n(\rho_{\mathrm{in}}))$$

Challenge: find the noisy basis $\mathcal{B}_n$.

Probabilistic error cancellation

Finding noisy basis

- Use a tree-combination of small noises
- Propagate all possible combination and sample them according to probability distribution given by α_n
- Not exhaustive, but average are good enough

$$\mathcal{E}_{\rm unit} = (1-p)\mathcal{U} + p\mathcal{N} \rightarrow$$

$$\mathcal{U} = \frac{1}{1-p}\mathcal{E}_{\rm unit} - \frac{p}{1-p}\mathcal{N}$$

Over all possible faults we choose $\mathcal{E}$ and $\mathcal{N}$, and sample them accordingly to the probability
Assuming all p_s are equal we obtain a sampling overhead

$$C_{\rm em} = \prod_{m=1}^{M}\left(\frac{1+p}{1-p}\right)^2 \approx e^{4\lambda}$$

Probabilistic error cancellation

- The sampling overhead depends dramatically in the choice of basis
- For combining many small errors, one can use gate set tomography
- There exist successful implementations in correlated noise
- Extensible to measurement errors
- Not clear how good this model captures emergent properties

Symmetry verification

- In some cases, the quantum computation satisfies some symmetry
- We can post-select and discard outputs with broken symmetries

Examples of symmetries: parity, spin, particle number... Given a physical system dominated by the Hamiltonian H, its symmetries are given by S such that

$$[H, S] = 0,$$

and thus S and H can be diagonalized simultaneously.

Thus, if we know that the output of a computation is such that $S|\psi\rangle = s|\psi\rangle$, we can measure S on the output and check the value s.

State symmetrization

Let Π be the projector on the s-eigenspace of the symmetry S. Then

$$\rho_{\mathrm{sym}} = \frac{\Pi\rho\Pi}{\mathrm{Tr}(\Pi\rho\Pi)}$$

and equivalently

$$\mathrm{Tr}(O\rho_{\mathrm{sym}}) = \frac{\mathrm{Tr}(\rho O_{\mathrm{sym}})}{\mathrm{Tr}(\Pi\rho)}$$

Symmetry verification

We can choose between directly measuring the symmetries or measuring other simplified circuits and post-process

Post-selection

- Only acceptance (or not) of a measured outcome
- The number of valid outcomes decreases, so the variance increases
- Modest sampling overhead

$$C_{\mathrm{em}} = \mathrm{Tr}(\rho \Pi)^{-1}$$

(with possibly sophisticated measurements)

Post-processing

- Decompose the symmetrized observable in smaller pieces
- Sample those smaller pieces according to Monte Carlo
- Number of valid outcomes decreases for symmetry, plus Monte Carlo uncertainty

$$C_{\mathrm{em}} = \mathrm{Tr}(\rho \Pi)^{-2}$$

(with simpler measurements)

Symmetry verification

- Symmetries are particularly useful in physically motivated or chemistry problems, since the provide huge simplification
- Finding symmetries is difficult, but there exist interesting transformations (e.g. fermion-to-qubit mappings) at hand
- We do not have to restrict ourselves to the existing symmetries, we can enforce them
  - In fact this is the key element of Quantum Error Correction

Purity-based error mitigation

- Noise channels decrease the purity of the states
- Virtual distillation aims to collectively measure ρ to access expectation values of the purified states

For

$$\rho = \sum_i p_i \left|\phi_i\right\rangle \left\langle\phi_i\right|$$

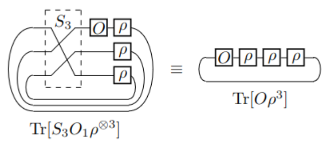
then

$$\rho_{\mathrm{pur}}^{(M)} = \frac{\rho^M}{\mathrm{Tr}(\rho^M)} = \frac{1}{\sum_i p_i^M} \sum_i p_i^M \left|\phi_i\right\rangle \left\langle\phi_i\right|$$

which effectively suppresses the states with small p_i

Virtual distillation

Computing $\mathrm{Tr}\left(O\rho^M\right)$ adding symmetric swaps[a]



[a] These methods require global measurements, which can be challenging. However, there exist techniques to reduce these requirements.

$$C_{em} = \mathrm{Tr}\left(\rho^M\right)^{-1} \approx e^{-M\lambda}$$

Purity-based error mitigation

- Noise channels decrease the purity of the states
- Virtual distillation aims to collectively measure ρ to access expectation values of the purified states

For

$$\rho = \sum_i p_i \left|\phi_i\right\rangle \left\langle\phi_i\right|$$

then

$$\rho_{\mathrm{pur}}^{(M)} = \frac{\rho^M}{\mathrm{Tr}(\rho^M)} = \frac{1}{\sum_i p_i^M} \sum_i p_i^M \left|\phi_i\right\rangle \left\langle\phi_i\right|$$

which effectively suppresses the states with small p_i

Echo verification

- In echo verification, we aim to project over the noiseless state
- We can do it by applying the quantum computation backwards
- The state is given by $\rho\Pi\rho\Pi + \Pi\rho\Pi\rho$

Similar to virtual distillation with $M = 2$

$$C_{em} = \mathrm{Tr}\left(\rho^M\right)^{-1} \approx e^{-2\lambda}$$

Overview

Methods	Probabilistic Error Cancellation	Richardson Extrapolation (Equal-gap#)	Symmetry Verification	Virtual Distillation	Echo Verification
Main assumptions	Full knowledge of the noise.	Ability to scale the noise. Small λ.¶	The ideal state contains inherent symmetry.	The ideal state ρ_0 is pure. The noise is stochastic such that ρ_0 is the dominant eigenvector of the noisy state ρ. §	
Hyper-parameters	Circuit fault rate *after* mitigation: λ_{em}	Number of data points: M	The projector of the symmetry subspace: Π	Degree of purification: M	Nil
Qubit overhead	1	1	1‡	M	1
Circuit runtime overhead	up to ~ 2	up to $\sim M$	1	~ 1‖	2
Sampling overhead (C_{em})	$e^{4(\lambda-\lambda_{em})}$	$(2^M-1)^2$	Post-selection: $\mathrm{Tr}[\Pi\rho]^{-1}$ Post-processing: $\mathrm{Tr}[\Pi\rho]^{-2}$	$\mathrm{Tr}\left[\rho^M\right]^{-2} \gtrsim \frac{e^{2M\lambda}}{\left[1+(e^{\lambda}-1)^M\right]^2}$	$\mathrm{Tr}\left[\rho^2\right]^{-1} \gtrsim \frac{e^{2\lambda}}{1+(e^{\lambda}-1)^2}$
Fidelity boost*	$e^{\lambda-\lambda_{em}}$	$e^{\lambda}+\mathcal{O}(\lambda^M)$	$\mathrm{Tr}[\Pi\rho]^{-1}$	$\frac{\mathrm{Tr}[\rho_0\rho]^{M-1}}{\mathrm{Tr}[\rho^M]} \gtrsim \frac{e^{\lambda}}{1+(e^{\lambda}-1)^M}$	Same as VD with $M=2$
Bias	Can reach 0 when $\lambda_{em}=0$.	$\mathcal{O}(\lambda^M)$	Can be upper-bounded using the error-mitigated fidelity using Eq. (53), which in turns is related to the fidelity boost achieved.		

General comments

- There exist other techniques to deal with errors
  - Subspace expansion
  - N-representability (cheap representations taking advantages of symmetries)
  - Machine-learning errors
- The methods can be combined for better performance
  - If combined in parallel, different methods can be used in different aspects (evolution, measurement...)
  - If combined sequentially, one method may destroy the error characterization for the next one

Section 4

Quantum Error Correction

Quantum Error Correction

Final goal for fault-tolerant quantum computing

- Logical qubits are encoded accross several physical qubits
- The physical qubits are measured to detect syndromes, that map to errors
- When an error is detected, it is corrected on the fly

Requirements

- Constantly measuring the circuit
- Constantly applying corrections to the state
- The step syndromes $\rightarrow$errors is sometimes difficult
- Maximal error threshold

Threshold theorem

Threshold theorem for quantum computation: A quantum circuit containing $p(n)$ gates may be simulated with probability of error at most ϵ using

$$O(\text{poly}(\log p(n)/\epsilon)p(n)) \qquad (10.116)$$

gates on hardware whose components fail with probability at most p, provided p is below some constant *threshold*, $p < p_{\text{th}}$, and given reasonable assumptions about the noise in the underlying hardware.

- No quantum error correction is available until we reach the thresholds
- Hardware requirements are larger

- Error correction is capable of suppressing all errors
- Error mitigation is a cheaper alternative, for the moment

5-qubit code

A flavour of quantum error correction

Measurements: $$\begin{aligned} M_1 &= XZZXI \\ M_2 &= IXZZX \\ M_3 &= XIXZZ \\ M_4 &= ZXIXZ \end{aligned}$$

$$|\bar{0}\rangle = \sum_{i,j} M_i M_j |00000\rangle$$
$$|\bar{1}\rangle = \sum_{i,j} M_i M_j |11111\rangle$$

A syndrome occurs when an error happens, and we detect it by measuring all M

$$\mathcal{E} = IIXII = X_3 \rightarrow \begin{aligned} \{\mathcal{E}, M_1\} &= 0 \\ \{\mathcal{E}, M_2\} &= 0 \\ [\mathcal{E}, M_3] &= 0 \\ [\mathcal{E}, M_4] &= 0 \end{aligned}$$

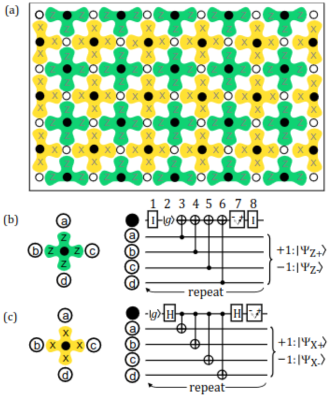
Syndrome table

	M_1	M_2	M_3	M_4
X_1				x
Z_1	x		x	
X_2	x			
Z_2		x		x
X_3	x	x		
Z_3			x	
X_4		x	x	
Z_4	x			x
X_5			x	x
Z_5		x		

x $\rightarrow \{,\} = 0$, blank $\rightarrow [,] = 0$

Surface code

- Scalable approach to fault-tolerant quantum computing
- The control measurements and syndromes extend for the entire circuits
- We need to decode syndromes to infer the errors



Section 5

Conclusions

Conclusions

- Error mitigation (EM) is a technique to improve results available today
- EM must come with some overhead, and this overhead is exponential in some resources, e.g. shots or effort in measurements
- EM corrects noise on average, but cannot correct specific errors
- Several techniques exist for EM, choose yours according to the task of interest
- Does not critically depend on any threshold, but the resource demands scale with error
- Intermediate step towards fault-tolerant quantum computing, which needs quantum error correction (EC)
- EC is more demanding than EM, but comes with different overheads

References

- Cai, Zhenyu et al. *Quantum Error Mitigation*. Dec. 2023. DOI: 10.48550/arXiv.2210.00921. arXiv: 2210.00921 [quant-ph].
- Fowler, Austin G. et al. "Surface Codes: Towards Practical Large-Scale Quantum Computation". In: *Physical Review A* 86.3 (Sept. 2012), p. 032324. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.86.032324. arXiv: 1208.0928 [quant-ph].
- Gottesman, Daniel. *The Heisenberg Representation of Quantum Computers*. July 1998. DOI: 10.48550/arXiv.quant-ph/9807006. arXiv: quant-ph/9807006.
- Nielsen, Michael A. and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge: Cambridge University Press, 2010. ISBN: 978-1-107-00217-3. DOI: 10.1017/CBO9780511976667.

Error mitigation

Adrián Pérez Salinas

[1]Lorentz Institute - Leiden University
[2]$\langle aQa \rangle^L$: Applied Quantum Algorithms

Spring school in near-term quantum computing
— Benasque 2024 —