Noise-aware mixed state quantum computation and its applications

Giuseppe Clemente
(giuseppe.clemente@unipi.it)
collaborator: Kevin Zambello

Istituto Nazionale di Fisica Nucleare

IN SUPREMÆ DIGNITATIS 1343
UNIVERSITÀ DI PISA

29 July 2024, Liverpool
41st Lattice Conference 2024

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
UNIVERSITÀ DI PISA

Introduction

To date, we are probably still far from realizing quantum advantage over classical algorithms. NISQ results cannot rely on ideally noiseless circuits.

Introduction

To date, we are probably still far from realizing quantum advantage over classical algorithms. NISQ results cannot rely on ideally noiseless circuits.

However, **unitary circuits**, never exactly realizable on real hardware, are currently used to model most of the QC systems and algorithms in the community, from real-time evolution to ground state computations.

Introduction

To date, we are probably still far from realizing quantum advantage over classical algorithms. NISQ results cannot rely on ideally noiseless circuits.

However, **unitary circuits**, never exactly realizable on real hardware, are currently used to model most of the QC systems and algorithms in the community, from real-time evolution to ground state computations.

We stress the need for a perspective shift for computations in the NISQ era:

replacing (unitary) circuits by non-unitary protocols, $U \rightarrow \mathcal{E}$

Introduction

To date, we are probably still far from realizing quantum advantage over classical algorithms. NISQ results cannot rely on ideally noiseless circuits.

However, **unitary circuits**, never exactly realizable on real hardware, are currently used to model most of the QC systems and algorithms in the community, from real-time evolution to ground state computations.

We stress the need for a perspective shift for computations in the NISQ era:

replacing (unitary) circuits by non-unitary protocols, $U \to \mathcal{E}$

Even with fault-tolerance, some problems involve a mixed state preparation/evolution (for example for thermal states or for modeling open systems).

From unitary to non-unitary channels

Ideally

Unitary circuit: pure to pure

$$|\psi\rangle \to |\psi'\rangle = U|\psi\rangle$$

Realistically

Quantum channel: mixed/pure to mixed

$$\rho \to \rho' = \mathcal{E}(\rho) = \sum_\alpha K_\alpha \rho K_\alpha^\dagger$$

In general, noisy hardware would make any ideal unitary operator into a quantum channel: $\mathcal{E}_U(\rho) \equiv U\rho U^\dagger \xrightarrow{noise} \widetilde{\mathcal{E}}_U(\rho)$

From unitary to non-unitary channels

Ideally

Unitary circuit: pure to pure

$$|\psi\rangle \rightarrow |\psi'\rangle = U|\psi\rangle$$

Realistically

Quantum channel: mixed/pure to mixed

$$\rho \rightarrow \rho' = \mathcal{E}(\rho) = \sum_{\alpha} K_\alpha \rho K_\alpha^\dagger$$

In general, noisy hardware would make any ideal unitary operator into a quantum channel: $\mathcal{E}_U(\rho) \equiv U\rho U^\dagger \xrightarrow{noise} \widetilde{\mathcal{E}}_U(\rho)$

While unitary operators form a group and admit inverse, quantum channels form a **semigroup** and cannot be generally inverted.
Pseudoinversion might still be possible: better results with smaller chunks.

From unitary to non-unitary channels

Ideally

Unitary circuit: pure to pure

$$|\psi\rangle \to |\psi'\rangle = U|\psi\rangle$$

Realistically

Quantum channel: mixed/pure to mixed

$$\rho \to \rho' = \mathcal{E}(\rho) = \sum_\alpha K_\alpha \rho K_\alpha^\dagger$$

In general, noisy hardware would make any ideal unitary operator into a quantum channel: $\mathcal{E}_U(\rho) \equiv U\rho U^\dagger \xrightarrow{noise} \widetilde{\mathcal{E}}_U(\rho)$

While unitary operators form a group and admit inverse, quantum channels form a **semigroup** and cannot be generally inverted.
Pseudoinversion might still be possible: better results with smaller chunks.

Quantum channels can also be engineered through partial measurements and/or stochastic sampling, not only noise.

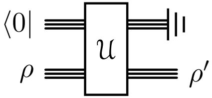
Some useful channel representations

One can engineer a channel through Stinespring representation or via statistical sampling of circuit ensembles.

Stinespring representation

Unitary on extended space $\mathcal{A} \otimes \mathcal{H}$ without measurement on ancilla register $\mathcal{A}$:

$$\rho \to \rho' = \mathrm{Tr}_{\mathcal{A}}[\mathcal{U}(|0\rangle\langle 0|_{\mathcal{A}} \otimes \rho)], \quad \mathcal{U} \in \mathrm{SU}(2^{q+q_a})$$

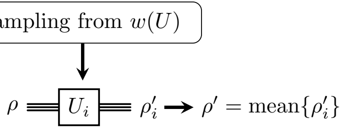


Stochastic sampling

Sample from circuits ensamble $\{U_i\} \sim w(U)$:

$$\rho \to \rho' \simeq \frac{1}{M} \sum_{i=1}^{M} U_i \rho U_i^\dagger$$

$$\rho' = \int_{\mathrm{SU}(2^q)} dU \; w(U) U \rho U^\dagger$$



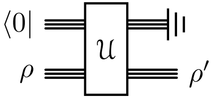
Some useful channel representations

One can engineer a channel through Stinespring representation or via statistical sampling of circuit ensembles.

Stinespring representation

Unitary on extended space $\mathcal{A} \otimes \mathcal{H}$ without measurement on ancilla register $\mathcal{A}$:

$$\rho \to \rho' = \mathrm{Tr}_{\mathcal{A}}[\mathcal{U}(|0\rangle\langle 0|_{\mathcal{A}} \otimes \rho)], \quad \mathcal{U} \in \mathrm{SU}(2^{q+q_a})$$

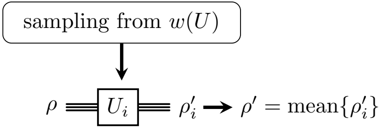


Stochastic sampling

Sample from circuits ensamble $\{U_i\} \sim w(U)$:

$$\rho \to \rho' \simeq \frac{1}{M} \sum_{i=1}^{M} U_i \rho U_i^\dagger$$

$$\rho' = \int_{\mathrm{SU}(2^q)} dU\, w(U) U \rho U^\dagger$$



In either representation, we can parameterize the protocol ($\mathcal{U}_{\vec{\theta}}$ or $w_{\vec{\theta}}(U)$).

Optimizing channels for unitary targets

Unitary task:
Assume we want to apply unitary operation $U \in \mathrm{SU}(2^q)$ on a noisy hardware.

Optimizing channels for unitary targets

Unitary task:
Assume we want to apply unitary operation $U \in \mathrm{SU}(2^q)$ on a noisy hardware.

Applying a specific decomposition of gates for U (or $\mathcal{E}_U$) actually realizes a noisy quantum channel $\widetilde{\mathcal{E}}_U$, not under control.

Optimizing channels for unitary targets

Unitary task:
Assume we want to apply unitary operation $U \in \mathrm{SU}(2^q)$ on a noisy hardware.

Applying a specific decomposition of gates for U (or $\mathcal{E}_U$) actually realizes a noisy quantum channel $\widetilde{\mathcal{E}}_U$, not under control.

We can engineer a parametric quantum channel $\widetilde{\mathcal{E}}_U^{(\vec{\theta})}$ where non-unitarity is partially under control.

Optimizing channels for unitary targets

Unitary task:
Assume we want to apply unitary operation $U \in \mathrm{SU}(2^q)$ on a noisy hardware.

Applying a specific decomposition of gates for U (or $\mathcal{E}_U$) actually realizes a noisy quantum channel $\widetilde{\mathcal{E}}_U$, not under control.

We can engineer a parametric quantum channel $\widetilde{\mathcal{E}}_U^{(\vec{\theta})}$ where non-unitarity is partially under control.

Our aim is then to minimize the difference between target and real parameterized channel

$$C(\vec{\theta}) \equiv d(\mathcal{E}_U, \widetilde{\mathcal{E}}_U^{(\vec{\theta})})$$

Proper distance between channels: the diamond norm

A proper definition of distance between two quantum channels $\mathcal{E}_1$, $\mathcal{E}_2$ is the **diamond norm** of their difference:

$$d_\diamond(\mathcal{E}_1, \mathcal{E}_2) \equiv \frac{1}{2} \sup_{n,\xi \geq 0} \operatorname{Tr} \left| (\mathcal{I}_n \otimes \mathcal{E}_1)(\xi) - (\mathcal{I}_n \otimes \mathcal{E}_2)(\xi) \right|.$$

Each channel $\mathcal{E}_i$ in $\mathcal{H}$ is *trivially* extended to an operator $\mathcal{I}_n \otimes \mathcal{E}_i$ in $\mathcal{A} \otimes \mathcal{H}$ ($n = \dim \mathcal{A}$), but states ξ can have *non-zero entanglement entropy* between $\mathcal{A}$ and $\mathcal{H}$ subsystems.

Proper distance between channels: the diamond norm

A proper definition of distance between two quantum channels $\mathcal{E}_1$, $\mathcal{E}_2$ is the **diamond norm** of their difference:

$$d_\diamond(\mathcal{E}_1, \mathcal{E}_2) \equiv \frac{1}{2} \sup_{n, \xi \geq 0} \mathrm{Tr} \left| (\mathcal{I}_n \otimes \mathcal{E}_1)(\xi) - (\mathcal{I}_n \otimes \mathcal{E}_2)(\xi) \right|.$$

Each channel $\mathcal{E}_i$ in $\mathcal{H}$ is *trivially* extended to an operator $\mathcal{I}_n \otimes \mathcal{E}_i$ in $\mathcal{A} \otimes \mathcal{H}$ ($n = \dim \mathcal{A}$), but states ξ can have *non-zero entanglement entropy* between $\mathcal{A}$ and $\mathcal{H}$ subsystems.

This extension is essential, since states on which an isolated quantum channel act are generally entangled with regions outside the channel.

Proper distance between channels: the diamond norm

A proper definition of distance between two quantum channels $\mathcal{E}_1$, $\mathcal{E}_2$ is the **diamond norm** of their difference:

$$d_\diamond(\mathcal{E}_1, \mathcal{E}_2) \equiv \frac{1}{2} \sup_{n, \xi \geq 0} \mathrm{Tr} \left| (\mathcal{I}_n \otimes \mathcal{E}_1)(\xi) - (\mathcal{I}_n \otimes \mathcal{E}_2)(\xi) \right|.$$

Each channel $\mathcal{E}_i$ in $\mathcal{H}$ is *trivially* extended to an operator $\mathcal{I}_n \otimes \mathcal{E}_i$ in $\mathcal{A} \otimes \mathcal{H}$ ($n = \dim \mathcal{A}$), but states ξ can have *non-zero entanglement entropy* between $\mathcal{A}$ and $\mathcal{H}$ subsystems.

This extension is essential, since states on which an isolated quantum channel act are generally entangled with regions outside the channel.

Fortunately, by *convexity*, $d_\diamond$ is saturated by ξ pure states
$\Longrightarrow$ possible preparation via unitaries: $\xi = V \left|0\right\rangle\!\left\langle 0\right| V^\dagger$. Trace distance can be estimated, e.g., via randomized measurements.

Known error mitigation techniques as particular cases

Mixed protocol can be implemented in different flavours, some specializing into already well known mitigation techniques:

- unitary mitigation channels as optimal encoders-decoders (e.g., **dynamical decoupling** in the case of identity maps);
- optimal sampling of ideally equivalent circuit ensembles (similar to **randomized compiling**);
- non-unitary mitigation channels as optimal channel encoders-decoders;
- non-unitary mitigation channel as embedded gate sampling;
- general circuits ensembles (e.g., optimal VQE ensembles of circuits as self-mitigating protocols).

Known error mitigation techniques as particular cases

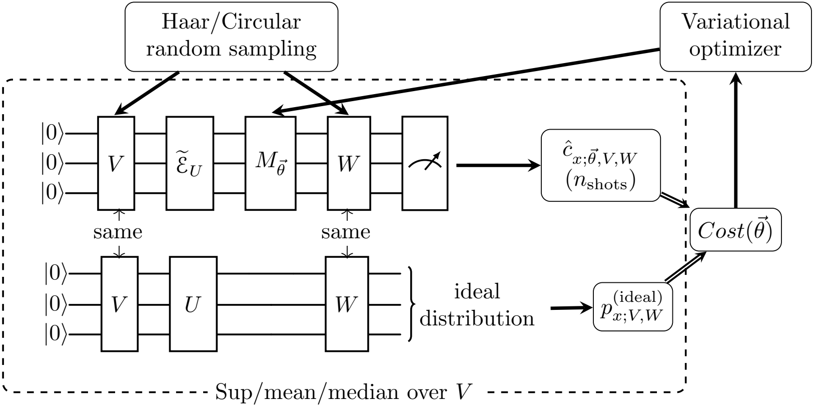
Mixed protocol can be implemented in different flavours, some specializing into already well known mitigation techniques:

- unitary mitigation channels as optimal encoders-decoders (e.g., **dynamical decoupling** in the case of identity maps);
- optimal sampling of ideally equivalent circuit ensembles (similar to **randomized compiling**);
- non-unitary mitigation channels as optimal channel encoders-decoders;
- non-unitary mitigation channel as embedded gate sampling;
- general circuits ensembles (e.g., optimal VQE ensembles of circuits as self-mitigating protocols).

The more the freedom in modeling $\widetilde{\mathcal{E}}^{(\vec{\theta})}$ the higher the training costs: some tradeoff is in order.

Simple example: unitary decoding

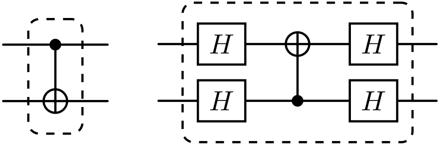
A single parameterized circuit $M_{\vec{\theta}}$ is trained to partially correct noise in $\widetilde{\mathcal{E}}_U$.



This also gives some explanation on how VQE algorithms are typically robust on noisy hardware.

Case study: stochastic CNOT with asymmetric noise

Let us consider $U = \mathrm{CNOT}$

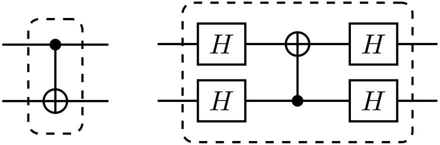


$\mathcal{E}_1 \xrightarrow{noise} \widetilde{\mathcal{E}}_1$ $\mathcal{E}_2 \xrightarrow{noise} \widetilde{\mathcal{E}}_2$

Without noise $\mathcal{E}_1 = \mathcal{E}_2 = \mathcal{E}_{\mathrm{CNOT}}$.
With asymmetric noise between the two qubits, the noisy version $\widetilde{\mathcal{E}}_i$ are different!

Case study: stochastic CNOT with asymmetric noise

Let us consider $U = \mathrm{CNOT}$



$\mathcal{E}_1 \xrightarrow{noise} \widetilde{\mathcal{E}}_1$ $\mathcal{E}_2 \xrightarrow{noise} \widetilde{\mathcal{E}}_2$

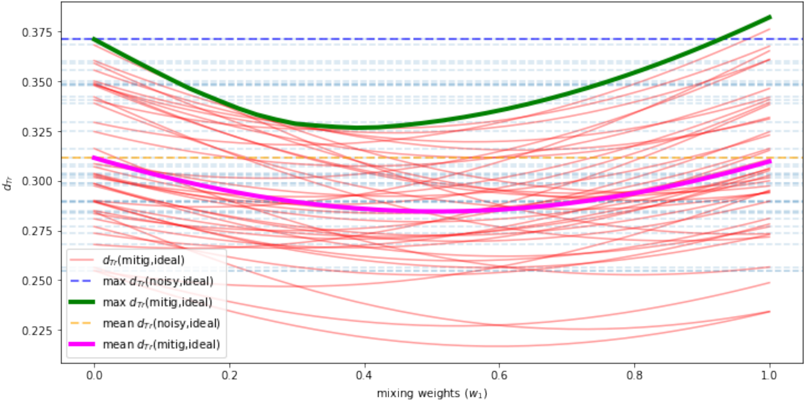
Without noise $\mathcal{E}_1 = \mathcal{E}_2 = \mathcal{E}_{\mathrm{CNOT}}$.
With asymmetric noise between the two qubits, the noisy version $\widetilde{\mathcal{E}}_i$ are different!

One can convexly mix the two channels with weights w_i:

$$\widetilde{\mathcal{E}}^{(\vec{w})} = w_1 \widetilde{\mathcal{E}}_1 + w_2 \widetilde{\mathcal{E}}_2 = w_1 \widetilde{\mathcal{E}}_1 + (1 - w_1) \widetilde{\mathcal{E}}_2.$$

Case study – stochastic CNOT with asymmetric noise: distance

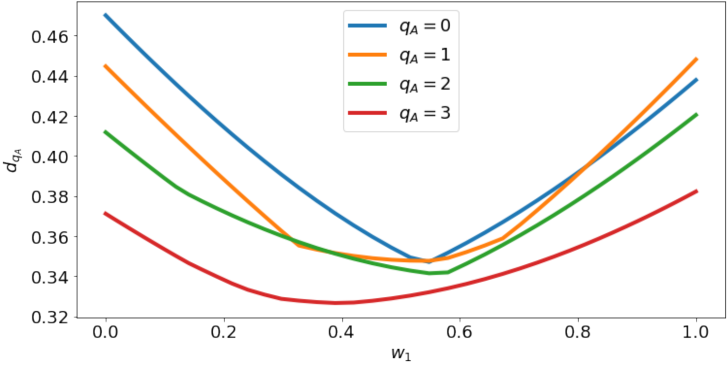
Cost as function of mixing weight w_1 for different input states



Minmaxing would select the best worst case $w_1^{(\text{opt})} \simeq 0.4$.

Case study – stochastic CNOT with asymmetric noise: extension dependence

Dependence on the extension dimension 2^{q_A} for diamond norm



In this case, no degradation on extension is observed.

Summary

Some main useful properties of mixed state quantum computation via parameterized quantum channels:

- generalizes unitary computing to NISQ era (give up ideal unitarity);
- it reduces to standard error mitigation techniques in some particular cases;
- can be realized in many different variants, depending on the task;
- it adapts to the specific noise properties of the QPU and gates/qubits involved.

Summary

Some main useful properties of mixed state quantum computation via parameterized quantum channels:

- generalizes unitary computing to NISQ era (give up ideal unitarity);
- it reduces to standard error mitigation techniques in some particular cases;
- can be realized in many different variants, depending on the task;
- it adapts to the specific noise properties of the QPU and gates/qubits involved.

Some open questions:

- trainability tradeoff and scaling?
- how crosstalks in real devices affects separate gate channels?

Summary

Some main useful properties of mixed state quantum computation via parameterized quantum channels:

- generalizes unitary computing to NISQ era (give up ideal unitarity);
- it reduces to standard error mitigation techniques in some particular cases;
- can be realized in many different variants, depending on the task;
- it adapts to the specific noise properties of the QPU and gates/qubits involved.

Some open questions:

- trainability tradeoff and scaling?
- how crosstalks in real devices affects separate gate channels?

Currently looking for <u>collaborators</u> for future investigations (anybody here?) and <u>access</u> to larger noisy hardware resources and also <u>feedbacks</u>.

Summary

Some main useful properties of mixed state quantum computation via parameterized quantum channels:

- generalizes unitary computing to NISQ era (give up ideal unitarity);
- it reduces to standard error mitigation techniques in some particular cases;
- can be realized in many different variants, depending on the task;
- it adapts to the specific noise properties of the QPU and gates/qubits involved.

Some open questions:

- trainability tradeoff and scaling?
- how crosstalks in real devices affects separate gate channels?

Currently looking for collaborators for future investigations (anybody here?) and access to larger noisy hardware resources and also feedbacks.

Thank you