Noise-assisted digital quantum simulation of open systems

José D. Guimarães, James Lim, Mikhail I. Vasilevskiy, Susana F. Huelga, Martin B. Plenio | 2022

Overview - Part I

- Use the intrinsic noise of NISQ devices as a resource for quantum computation.

Overview - Part I

- Use the intrinsic noise of NISQ devices as a resource for quantum computation.

- Quantum simulation of Markovian dynamics of open quantum systems;

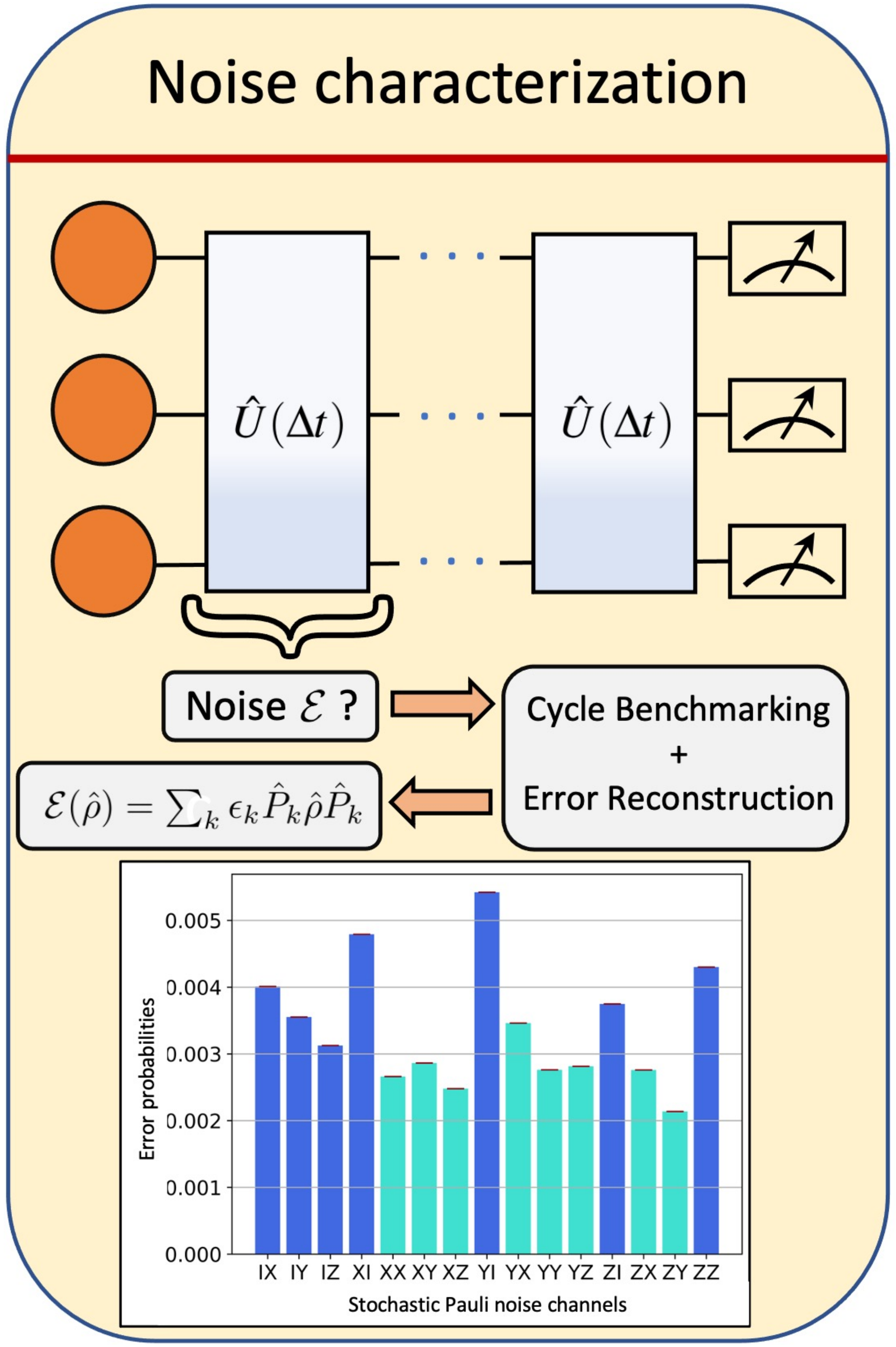
Overview - Part I

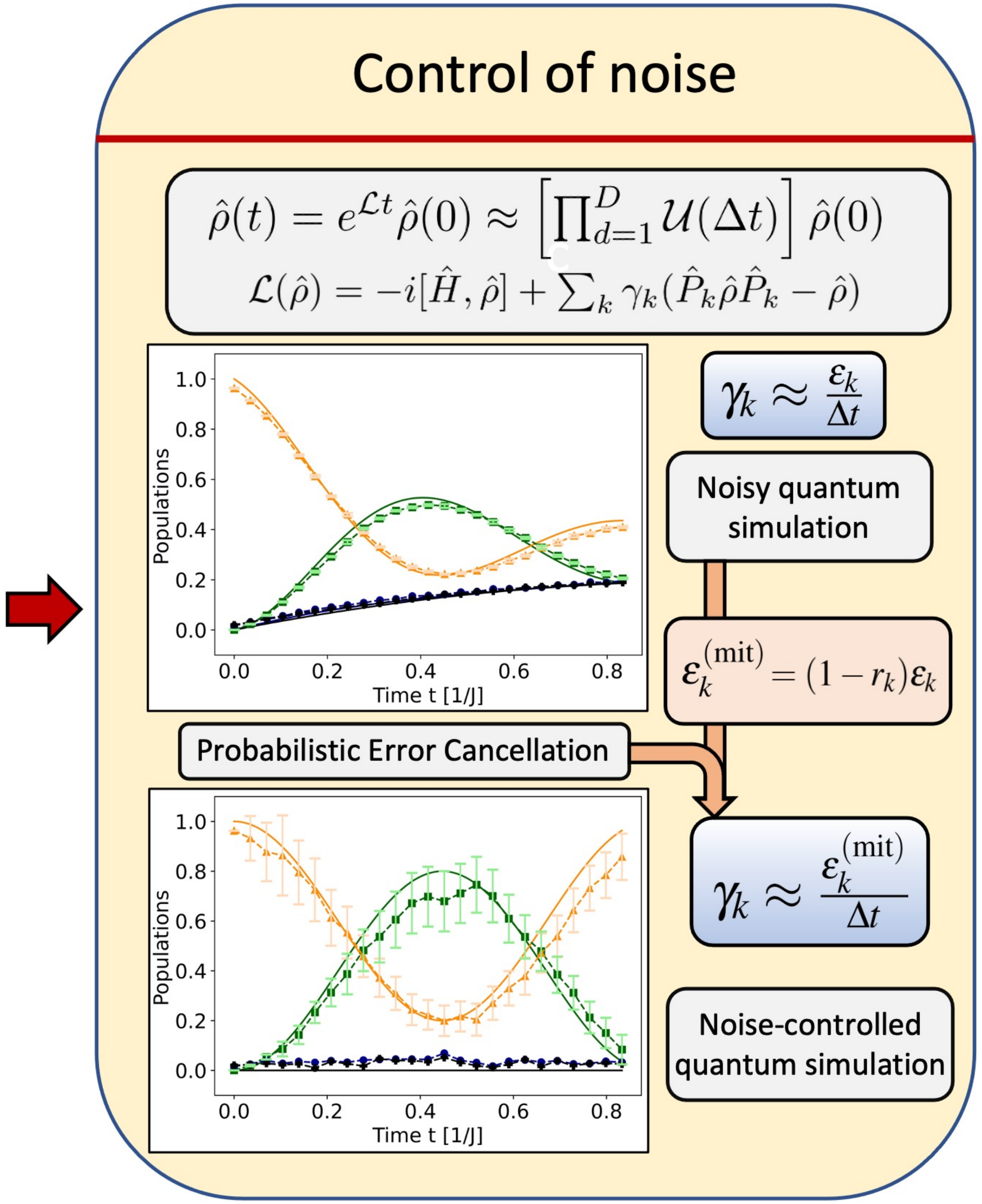
- Use the intrinsic noise of NISQ devices as a resource for quantum computation.

- Quantum simulation of Markovian dynamics of open quantum systems;

- The quantum hardware does not need to be changed in order to tune the decoherence rates;

Overview - Part II





Time-evolution via product formulas

- Time-evolution in the quantum computer,

Noiseless quantum computer: $e^{-i\hat{H}t}|\Psi(0)\rangle \approx \prod_{d=1}^{D} \hat{U}_k(\Delta t)|\Psi(0)\rangle,$ $\hat{U}_1(\Delta t) = \prod_{j=1}^{N} e^{-i\hat{H}_j\Delta t}, \quad \Delta t = t/D\,.$

$\hat{H} = \sum_{j=1}^{N} \hat{H}_j, \quad \hat{H}_j = \alpha_j \hat{P}_j$

Time-evolution via product formulas

- Time-evolution in the quantum computer,

Noiseless quantum computer: $e^{-i\hat{H}t}|\Psi(0)\rangle \approx \prod_{d=1}^{D} \hat{U}_k(\Delta t)|\Psi(0)\rangle, \qquad \hat{U}_1(\Delta t) = \prod_{j=1}^{N} e^{-i\hat{H}_j \Delta t}, \quad \Delta t = t/D\,.$

Noisy quantum computer: $\frac{d\hat{\rho}(t)}{dt} = \mathscr{L}[\hat{\rho}(t)] = -i[\hat{H}, \hat{\rho}(t)] + \mathscr{D}_{\text{intrinsic}}[\hat{\rho}(t)]$

- Markovian noise.
- Weak noise over a Trotter iteration.

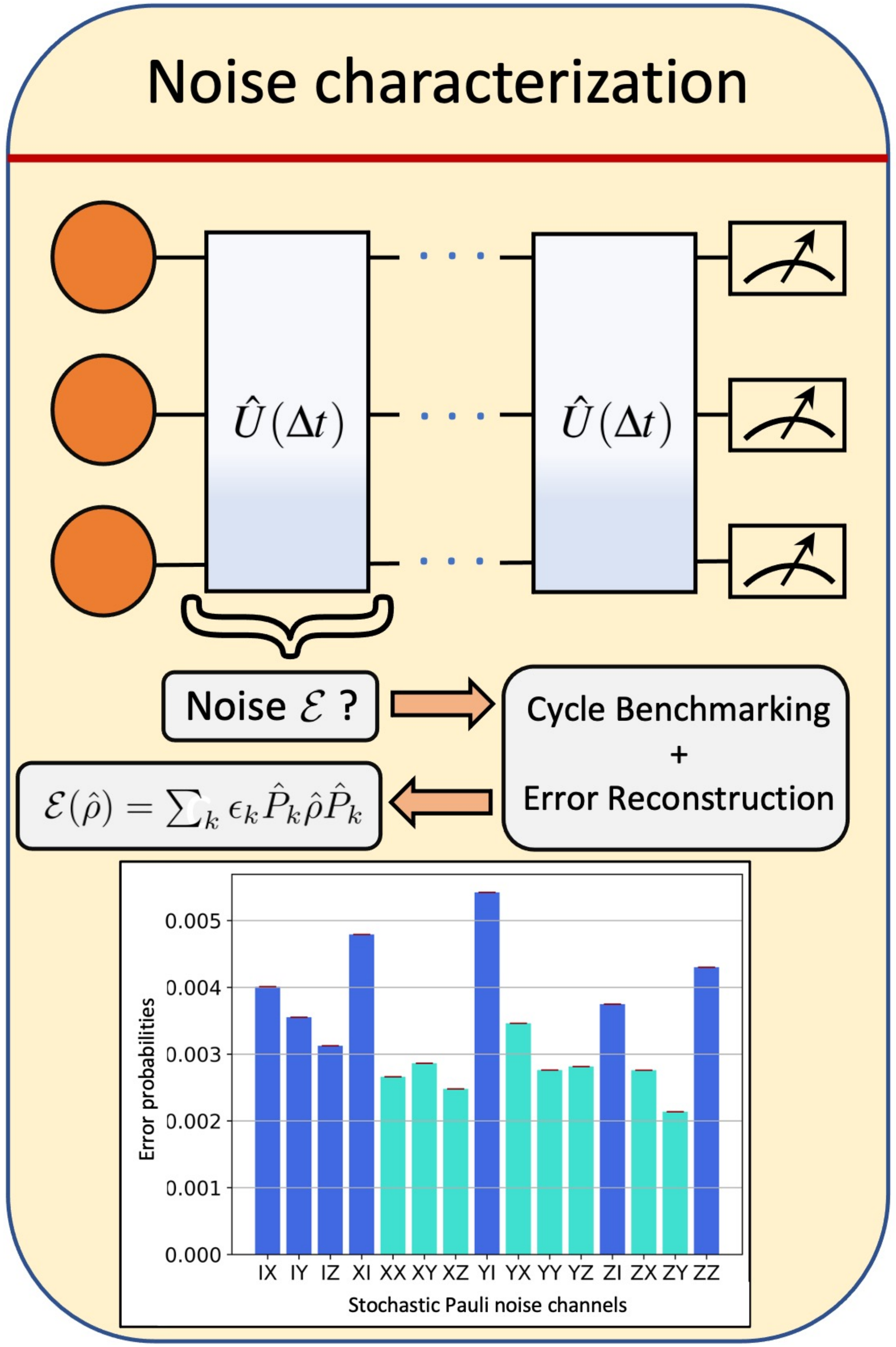
Noise characterization

CB + ER

Erhard, Alexander, et al. *Nature communications* 10.1 (2019): 1-7.

Flammia, Steven T., and Joel J. Wallman. *ACM Transactions on Quantum Computing* 1.1 (2020): 1-32.

K-qubit stochastic Pauli channel: $\mathscr{E}(\rho) = \sum_k \epsilon_k \hat{P}_k \rho \hat{P}_k$



Noise characterization

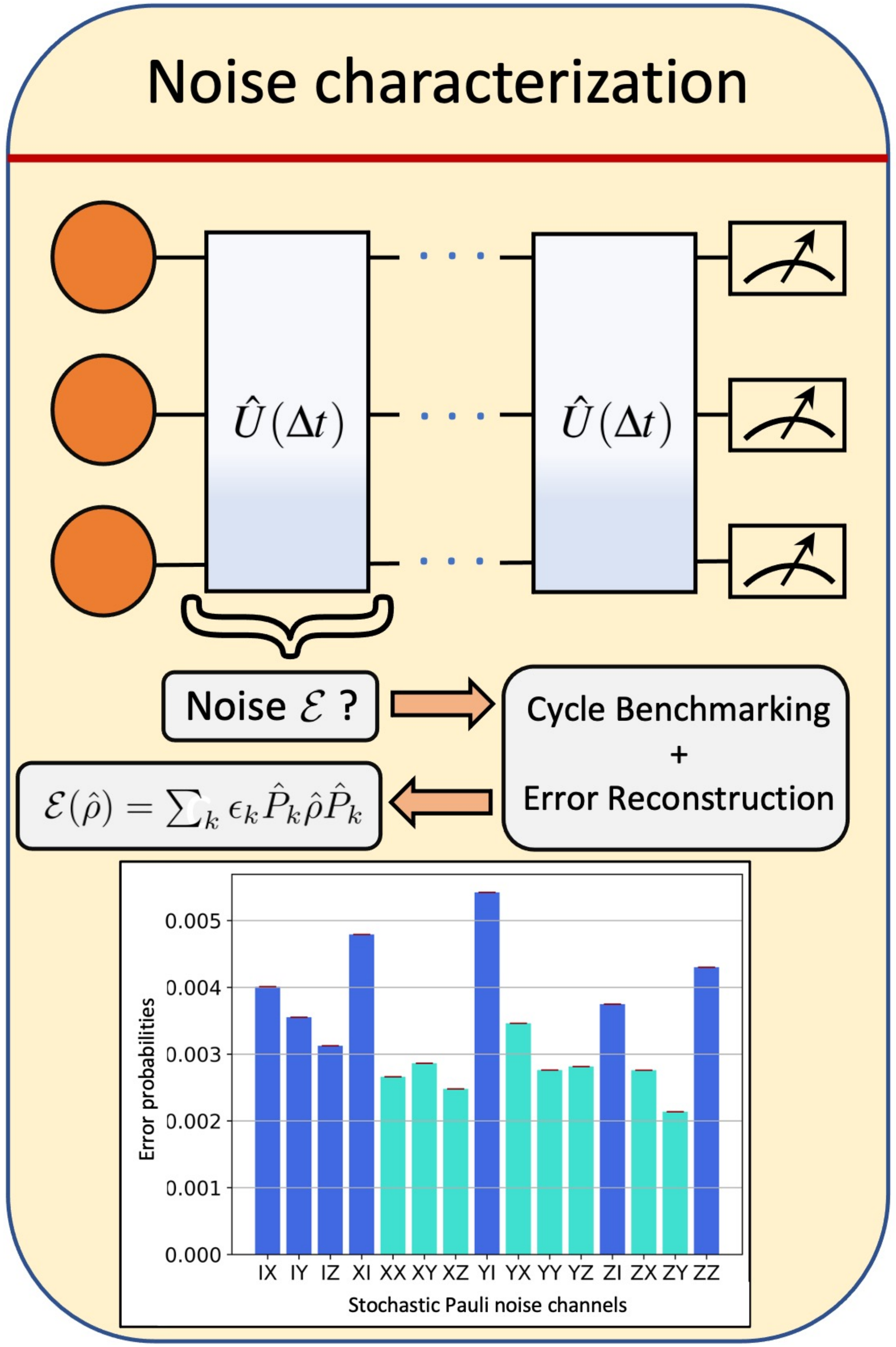
CB + ER

K-qubit stochastic Pauli channel: $\mathscr{E}(\rho) = \sum_k \epsilon_k \hat{P}_k \rho \hat{P}_k$

K=1: $\mathscr{E}_m^{(1)}(\hat{\rho}) = \epsilon_0 \hat{\rho} + \epsilon_X \hat{X}_m \hat{\rho} \hat{X}_m + \epsilon_Y \hat{Y}_m \hat{\rho} \hat{Y}_m + \epsilon_Z \hat{Z}_m \hat{\rho} \hat{Z}_m$

K=2: $\mathscr{E}_{m,m+1}^{(2)}(\hat{\rho}) = \mathscr{E}_m^{(1)}(\hat{\rho}) + \mathscr{E}_{m+1}^{(1)}(\hat{\rho})$

$+\epsilon_{XX} \hat{X}_m \hat{X}_{m+1} \hat{\rho} \hat{X}_m \hat{X}_{m+1}, + \epsilon_{XY} \hat{X}_m \hat{Y}_{m+1} \hat{\rho} \hat{X}_m \hat{Y}_{m+1} + \cdots$



Noise characterization

CB + ER

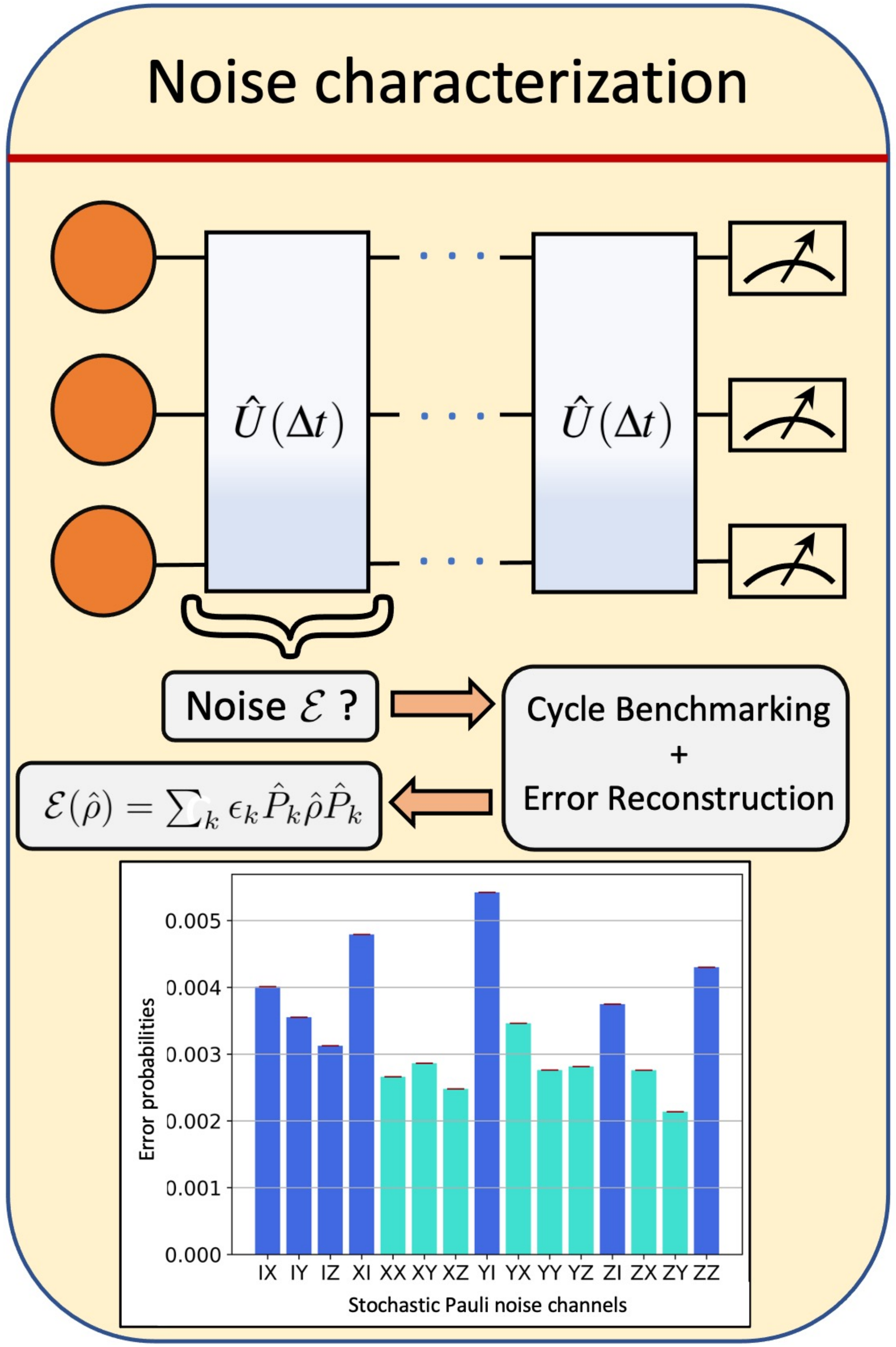
K-qubit stochastic Pauli channel: $\mathscr{E}(\rho) = \sum_k \epsilon_k \hat{P}_k \rho \hat{P}_k$

K=1: $\mathscr{E}_m^{(1)}(\hat{\rho}) = \epsilon_0 \hat{\rho} + \epsilon_X \hat{X}_m \hat{\rho} \hat{X}_m + \epsilon_Y \hat{Y}_m \hat{\rho} \hat{Y}_m + \epsilon_Z \hat{Z}_m \hat{\rho} \hat{Z}_m$

K=2: $\mathscr{E}_{m,m+1}^{(2)}(\hat{\rho}) = \mathscr{E}_m^{(1)}(\hat{\rho}) + \mathscr{E}_{m+1}^{(1)}(\hat{\rho})$

$$+\epsilon_{XX} \hat{X}_m \hat{X}_{m+1} \hat{\rho} \hat{X}_m \hat{X}_{m+1}, + \epsilon_{XY} \hat{X}_m \hat{Y}_{m+1} \hat{\rho} \hat{X}_m \hat{Y}_{m+1} + \cdots$$

No need to characterize errors acting on more than K=2 nearest-neighbour qubits.



Hashim, Akel, et al. *arXiv preprint arXiv:2010.00215* (2020).
Wallman, Joel J., and Joseph Emerson. *Physical Review A* 94.5 (2016): 052325.

Noise characterization

CB + ER

K-qubit stochastic Pauli channel: $\mathscr{E}(\rho) = \sum_k \epsilon_k \hat{P}_k \rho \hat{P}_k$

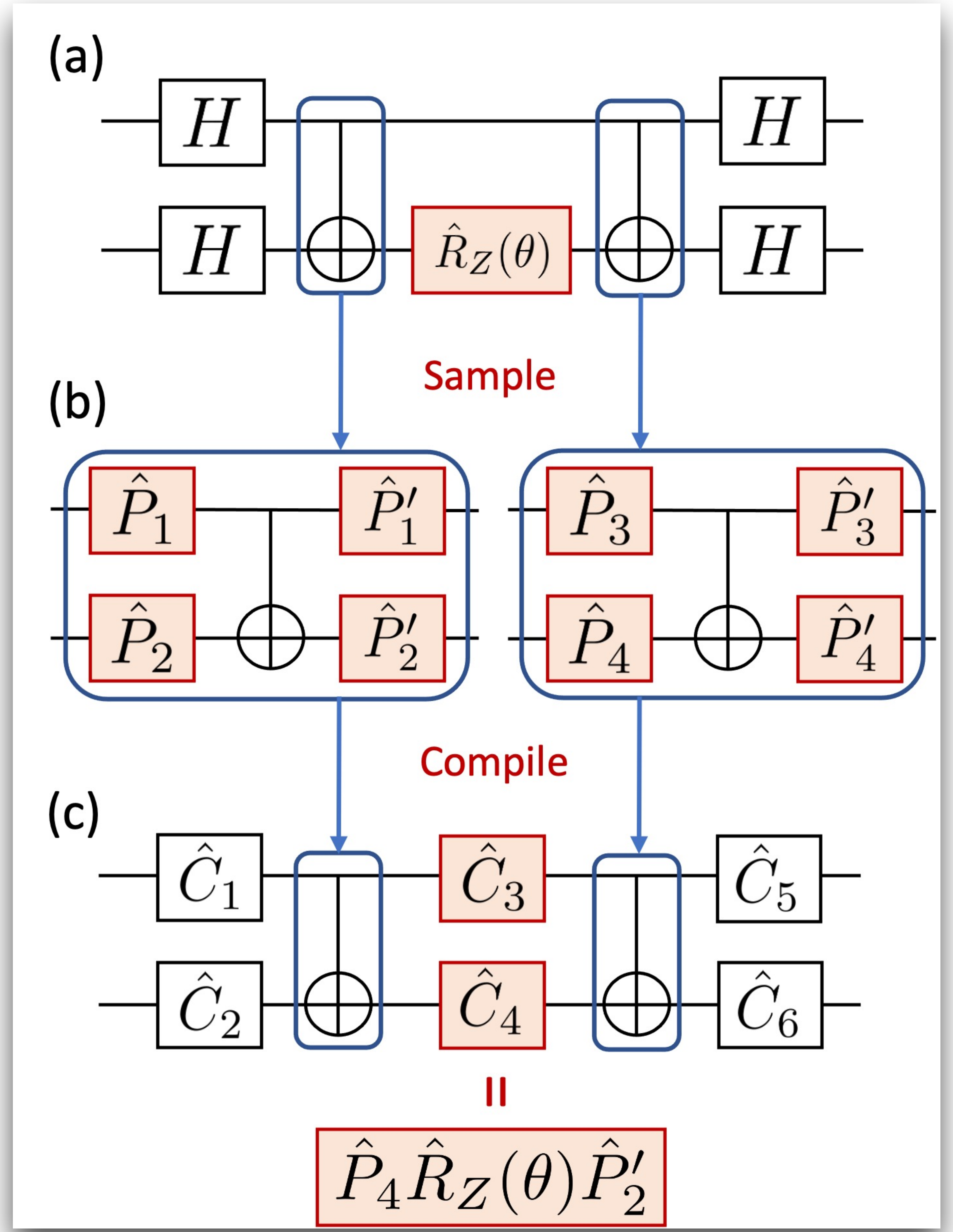
K=1: $\mathscr{E}_m^{(1)}(\hat{\rho}) = \epsilon_0 \hat{\rho} + \epsilon_X \hat{X}_m \hat{\rho} \hat{X}_m + \epsilon_Y \hat{Y}_m \hat{\rho} \hat{Y}_m + \epsilon_Z \hat{Z}_m \hat{\rho} \hat{Z}_m$

K=2: $\mathscr{E}_{m,m+1}^{(2)}(\hat{\rho}) = \mathscr{E}_m^{(1)}(\hat{\rho}) + \mathscr{E}_{m+1}^{(1)}(\hat{\rho})$

$+\epsilon_{XX} \hat{X}_m \hat{X}_{m+1} \hat{\rho} \hat{X}_m \hat{X}_{m+1}, + \epsilon_{XY} \hat{X}_m \hat{Y}_{m+1} \hat{\rho} \hat{X}_m \hat{Y}_{m+1} + \cdots$

No need to characterize errors acting on more than K=2 nearest-neighbour qubits.

Randomized Compiling



$\hat{G} = \hat{P}'_k \hat{G} \hat{P}_k, \quad \hat{P}'_k = \hat{G} \hat{P}_k \hat{G}^\dagger$

Time-evolution via product formulas

- Time-evolution in the quantum computer,

$$\hat{H} = \sum_{j=1}^{N} \hat{H}_j, \quad \hat{H}_j = \alpha_j \hat{P}_j$$

Noiseless quantum computer:

$$e^{-i\hat{H}t}|\Psi(0)\rangle \approx \prod_{d=1}^{D} \hat{U}_k(\Delta t)|\Psi(0)\rangle, \qquad \hat{U}_1(\Delta t) = \prod_{j=1}^{N} e^{-i\hat{H}_j \Delta t}, \quad \Delta t = t/D\,.$$

Noisy quantum computer:

$$\frac{d\hat{\rho}(t)}{dt} = \mathscr{L}[\hat{\rho}(t)] = -i[\hat{H}, \hat{\rho}(t)] + \mathscr{D}_{\text{stochastic}}[\hat{\rho}(t)]$$

- Markovian noise.
- Weak noise over a Trotter iteration.

Time-evolution via product formulas

$\hat{H} = \sum_{j=1}^{N} \hat{H}_j, \quad \hat{H}_j = \alpha_j \hat{P}_j$

- Time-evolution in the quantum computer,

Noiseless quantum computer: $e^{-i\hat{H}t}|\Psi(0)\rangle \approx \prod_{d=1}^{D} \hat{U}_k(\Delta t)|\Psi(0)\rangle,$ $\hat{U}_1(\Delta t) = \prod_{j=1}^{N} e^{-i\hat{H}_j\Delta t}, \quad \Delta t = t/D\,.$

Noisy quantum computer:

$$\frac{d\hat{\rho}(t)}{dt} = \mathscr{L}[\hat{\rho}(t)] = -i[\hat{H}, \hat{\rho}(t)] + \mathscr{D}_{\text{stochastic}}[\hat{\rho}(t)]$$

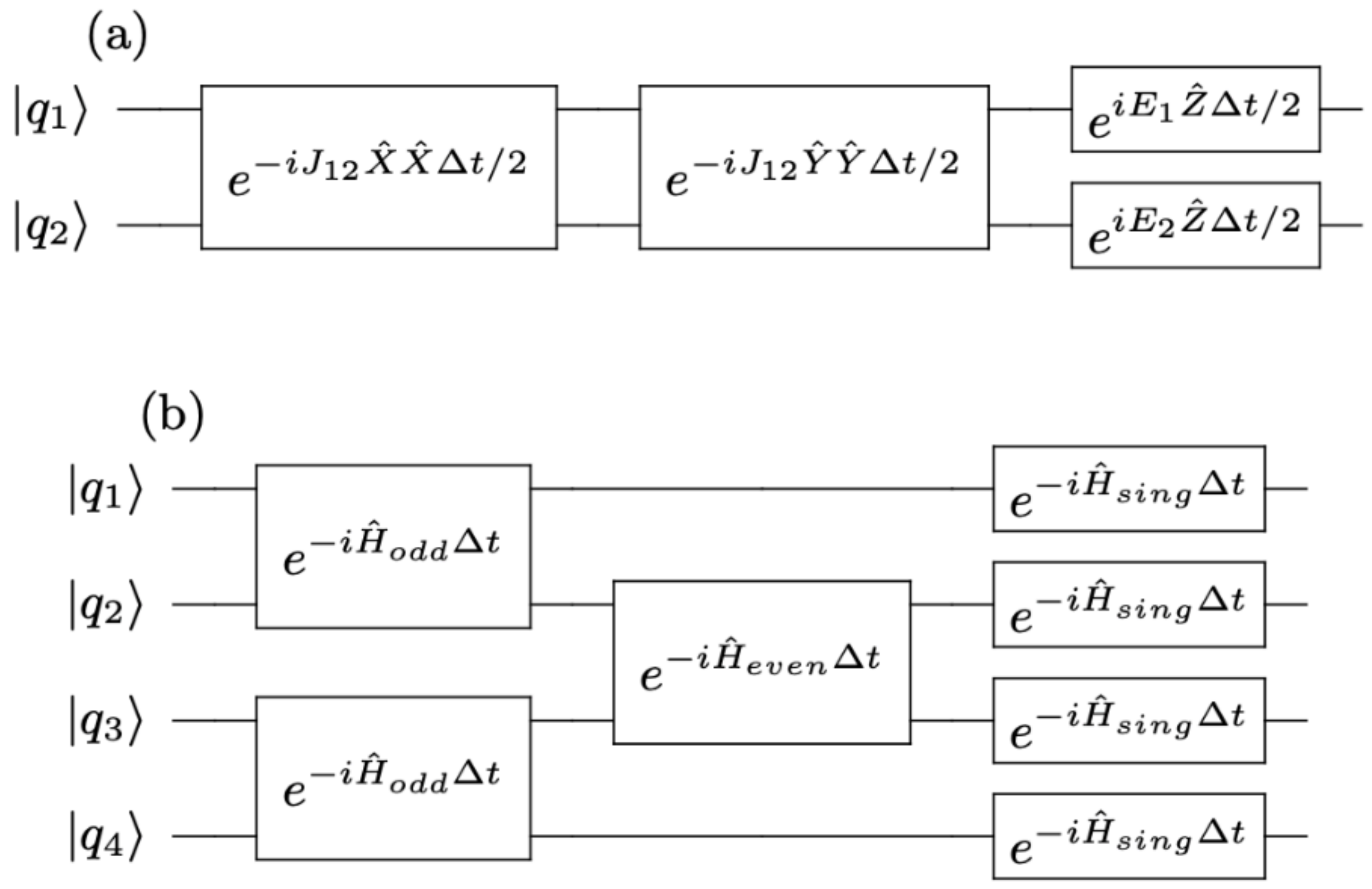
- Markovian noise.
- Weak noise over a Trotter iteration.

$$\mathscr{D}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t)\right), \quad \gamma_k = \epsilon_k/\Delta t$$

K=2

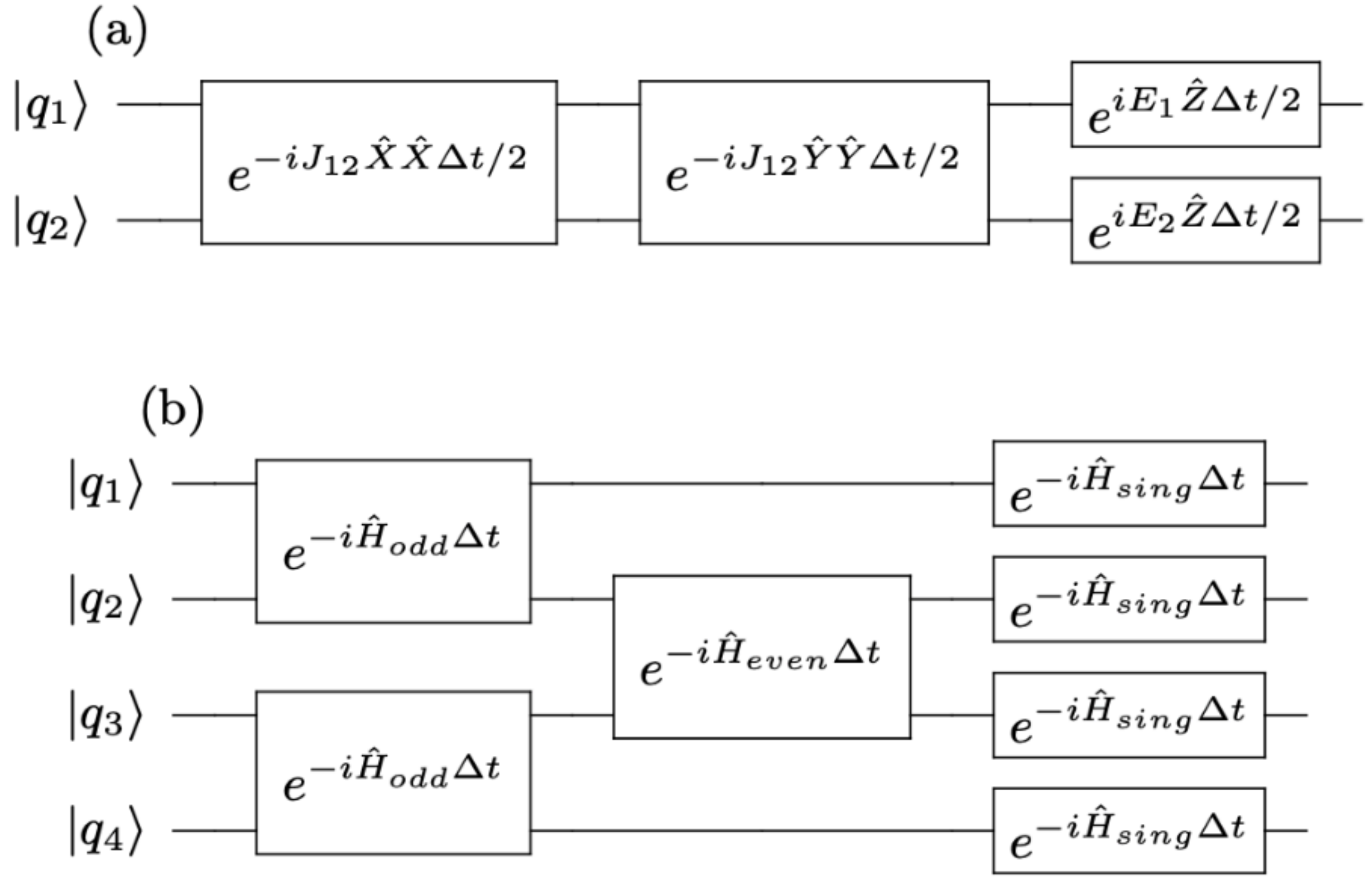
Time-evolution via product formulas

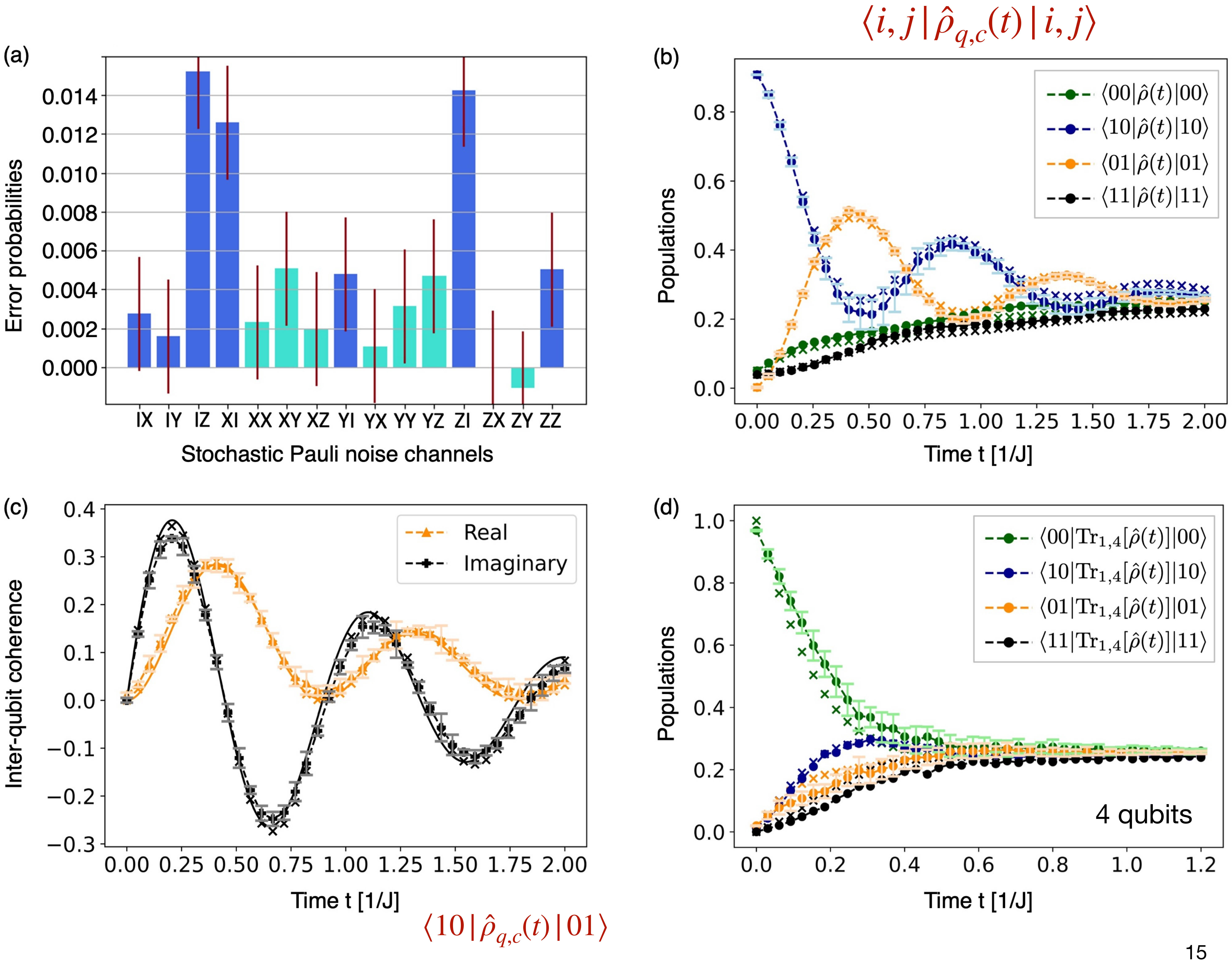
$$\hat{H} = -\sum_{m=1}^{n} \frac{E_m}{2}\hat{Z}_m + \sum_{m=1}^{n-1} \frac{J_{m,m+1}}{2}(\hat{X}_m\hat{X}_{m+1} + \hat{Y}_m\hat{Y}_{m+1})$$



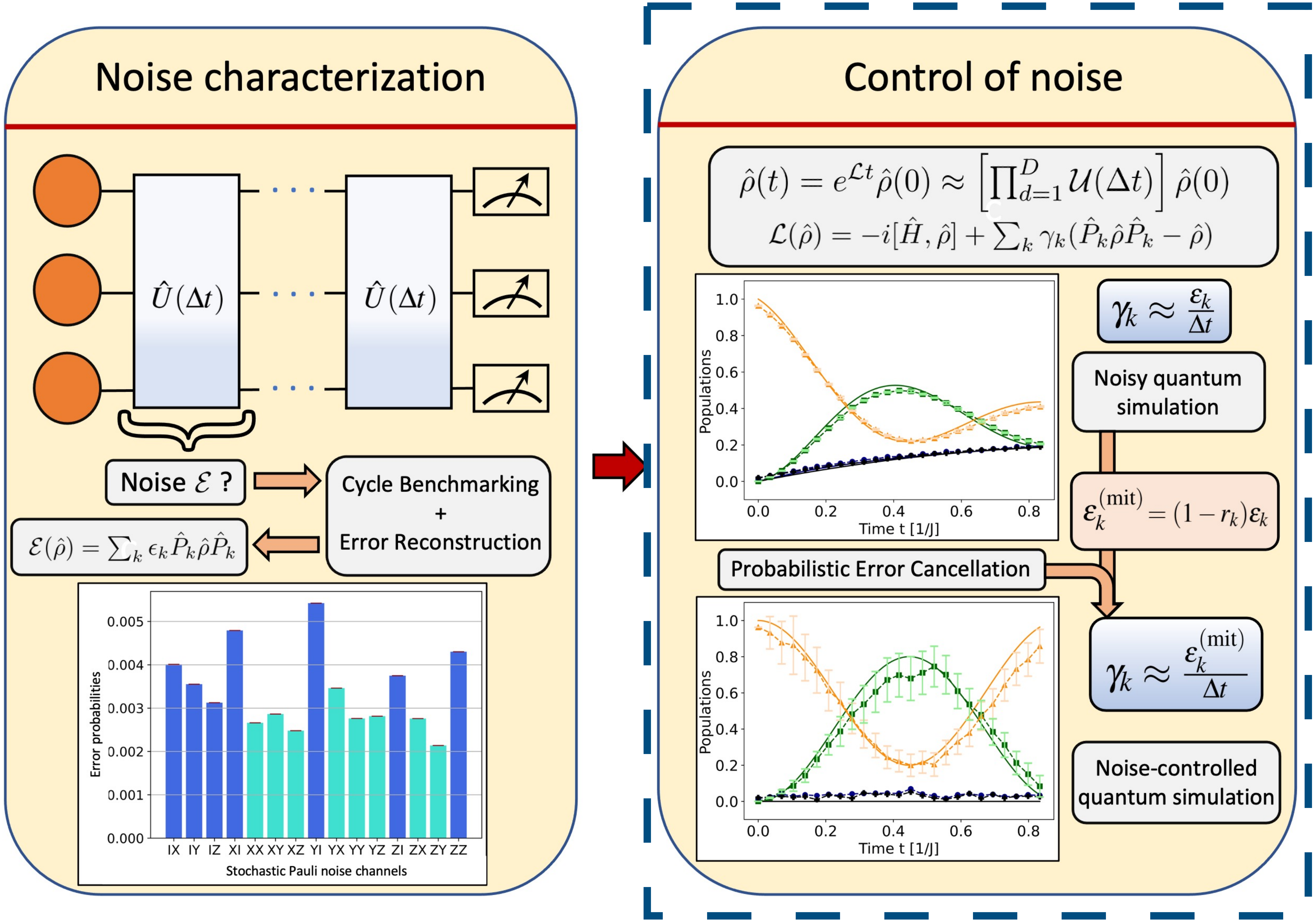
Time-evolution via product formulas

$$\hat{H} = -\sum_{m=1}^{n} \frac{E_m}{2}\hat{Z}_m + \sum_{m=1}^{n-1} \frac{J_{m,m+1}}{2}(\hat{X}_m\hat{X}_{m+1} + \hat{Y}_m\hat{Y}_{m+1})$$





Control of noise



Control of noise

$$\frac{d\hat{\rho}(t)}{dt} = \mathscr{L}[\hat{\rho}(t)] = -i[\hat{H}, \hat{\rho}(t)] + \mathscr{D}_{\text{stochastic}}[\hat{\rho}(t)]$$

$$\mathscr{D}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t)\right), \quad \gamma_k = \epsilon_k / \Delta t$$

How can we control γ_k in the digital simulation without changing the hardware components?

Control of noise

$$\frac{d\hat{\rho}(t)}{dt} = \mathcal{L}[\hat{\rho}(t)] = -i[\hat{H}, \hat{\rho}(t)] + \mathcal{D}_{\text{stochastic}}[\hat{\rho}(t)]$$

$$\mathcal{D}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t) \right), \quad \gamma_k = \epsilon_k / \Delta t$$

Target decoherence rates Γ_k

$$\mathcal{D}^{(\text{controlled})}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \Gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t) \right), \quad \Gamma_k = \epsilon_k (1 - r_k) / \Delta t\,.$$

$r_k \in [0,1]$

Probabilistic Error Cancelation

Endo, Suguru, Simon C. Benjamin, and Ying Li. *Physical Review X* 8.3 (2018): 031027.
Sun, Jinzhao, et al. *Physical Review Applied* 15.3 (2021): 034026.

Probabilistic Error Cancelation

Overview

- Probabilistically cancels Markovian noise to first order by inverting the noise channel.

Characterization of the noise channel acting in the circuit:

$$\mathscr{E}(\rho) = \epsilon_0 \rho + \sum_{k=1} \epsilon_k P_k \rho P_k$$

CPTP map

Inversion of the characterized stochastic Pauli channel:

$$\mathscr{E}^{-1}(\rho) = \epsilon_0' \rho - \sum_{k=1} \epsilon_k' P_k \rho P_k$$

Non-CP map

$$\epsilon_{k>0}' = r_k \epsilon_k$$

Probabilistic Error Cancelation

Overview

- Probabilistically cancels Markovian noise to first order by inverting the noise channel.

Characterization of the noise channel acting in the circuit:

$$\mathscr{E}(\rho) = \epsilon_0 \rho + \sum_{k=1} \epsilon_k P_k \rho P_k$$

CPTP map

Inversion of the characterized stochastic Pauli channel:

$$\mathscr{E}^{-1}(\rho) = \epsilon'_0 \rho - \sum_{k=1} \epsilon'_k P_k \rho P_k$$

Non-CP map

$$\epsilon'_{k>0} = r_k \epsilon_k$$

For each Trotter iteration apply

$$\mathscr{E}^{-1}(\rho) = \sum_{k=0} q_k \mathscr{P}_k(\rho) = \sum_{k=0} q_k P_k \rho P_k, \quad q_k = \pm \epsilon'_k$$

Probabilistic Error Cancelation

Overview

- Probabilistically cancels Markovian noise to first order by inverting the noise channel.

Characterization of the noise channel acting in the circuit: $\mathscr{E}(\rho) = \epsilon_0 \rho + \sum_{k=1} \epsilon_k P_k \rho P_k$ CPTP map

Inversion of the characterized stochastic Pauli channel: $\mathscr{E}^{-1}(\rho) = \epsilon'_0 \rho - \sum_{k=1} \epsilon'_k P_k \rho P_k$ Non-CP map

$\epsilon'_{k>0} = r_k \epsilon_k$

For each Trotter iteration apply

$$\mathscr{E}^{-1}(\rho) = \sum_{k=0} q_k \mathscr{P}_k(\rho) = \sum_{k=0} q_k P_k \rho P_k, \quad q_k = \pm\, \epsilon'_k$$

Probabilistic application of Pauli gates

$$\mathscr{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathscr{P}_k$$

$p_k = \epsilon'_k / C$

$C = \sum_{k=0} |q_k|$

Probabilistic Error Cancelation

Overview

- Probabilistically cancels Markovian noise to first order by inverting the noise channel.

Characterization of the noise channel acting in the circuit: $\mathscr{E}(\rho) = \epsilon_0 \rho + \sum_{k=1} \epsilon_k P_k \rho P_k$ CPTP map

Inversion of the characterized stochastic Pauli channel: $\mathscr{E}^{-1}(\rho) = \epsilon'_0 \rho - \sum_{k=1} \epsilon'_k P_k \rho P_k$ Non-CP map

For each Trotter iteration apply

$$\mathscr{E}^{-1}(\rho) = \sum_{k=0} q_k \mathscr{P}_k(\rho) = \sum_{k=0} q_k P_k \rho P_k, \quad q_k = \pm \epsilon'_k$$

Probabilistic application of Pauli gates

$$\mathscr{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathscr{P}_k$$

$$\epsilon'_{k>0} = r_k \epsilon_k$$

$$p_k = \epsilon'_k / C$$

Probability of sampling P_k

$$C = \sum_{k=0} |q_k|$$

Probabilistic Error Cancelation

Overview

- Probabilistically cancels Markovian noise to first order by inverting the noise channel.

Characterization of the noise channel acting in the circuit: $\mathscr{E}(\rho) = \epsilon_0 \rho + \sum_{k=1} \epsilon_k P_k \rho P_k$ CPTP map

Inversion of the characterized stochastic Pauli channel: $\mathscr{E}^{-1}(\rho) = \epsilon'_0 \rho - \sum_{k=1} \epsilon'_k P_k \rho P_k$ Non-CP map

$\epsilon'_{k>0} = r_k \epsilon_k$

For each Trotter iteration apply

$$\mathscr{E}^{-1}(\rho) = \sum_{k=0} q_k \mathscr{P}_k(\rho) = \sum_{k=0} q_k P_k \rho P_k, \quad q_k = \pm \epsilon'_k$$

Probabilistic application of Pauli gates

$$\mathscr{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathscr{P}_k$$

$p_k = \epsilon'_k / C$

$C = \sum_{k=0} |q_k|$ → Mitigation cost

Probabilistic Error Cancelation

Overview

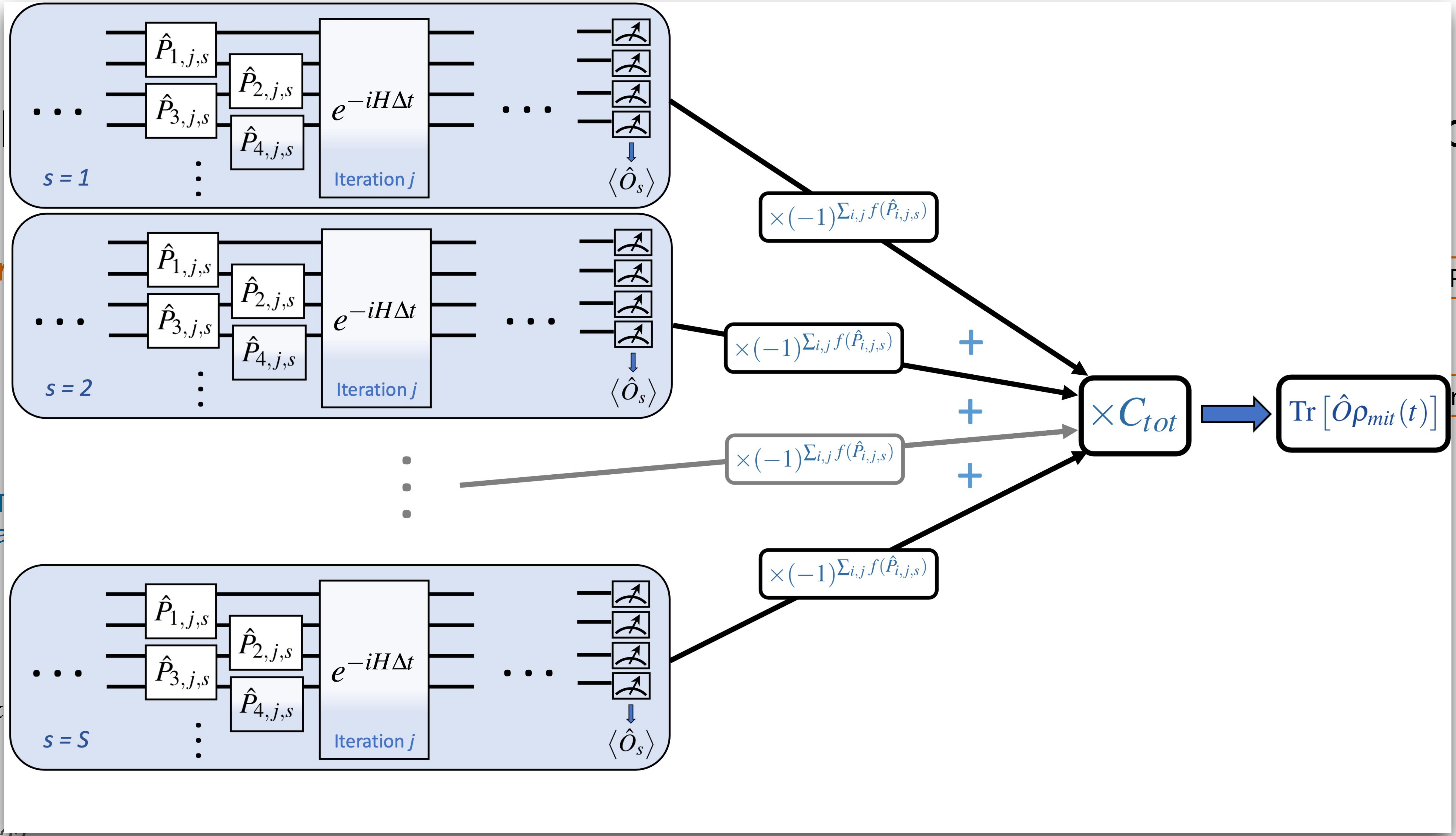
- Probabil ... oise channel

Character ... PTP map

Inversion ... n-CP map

For each T iteration a

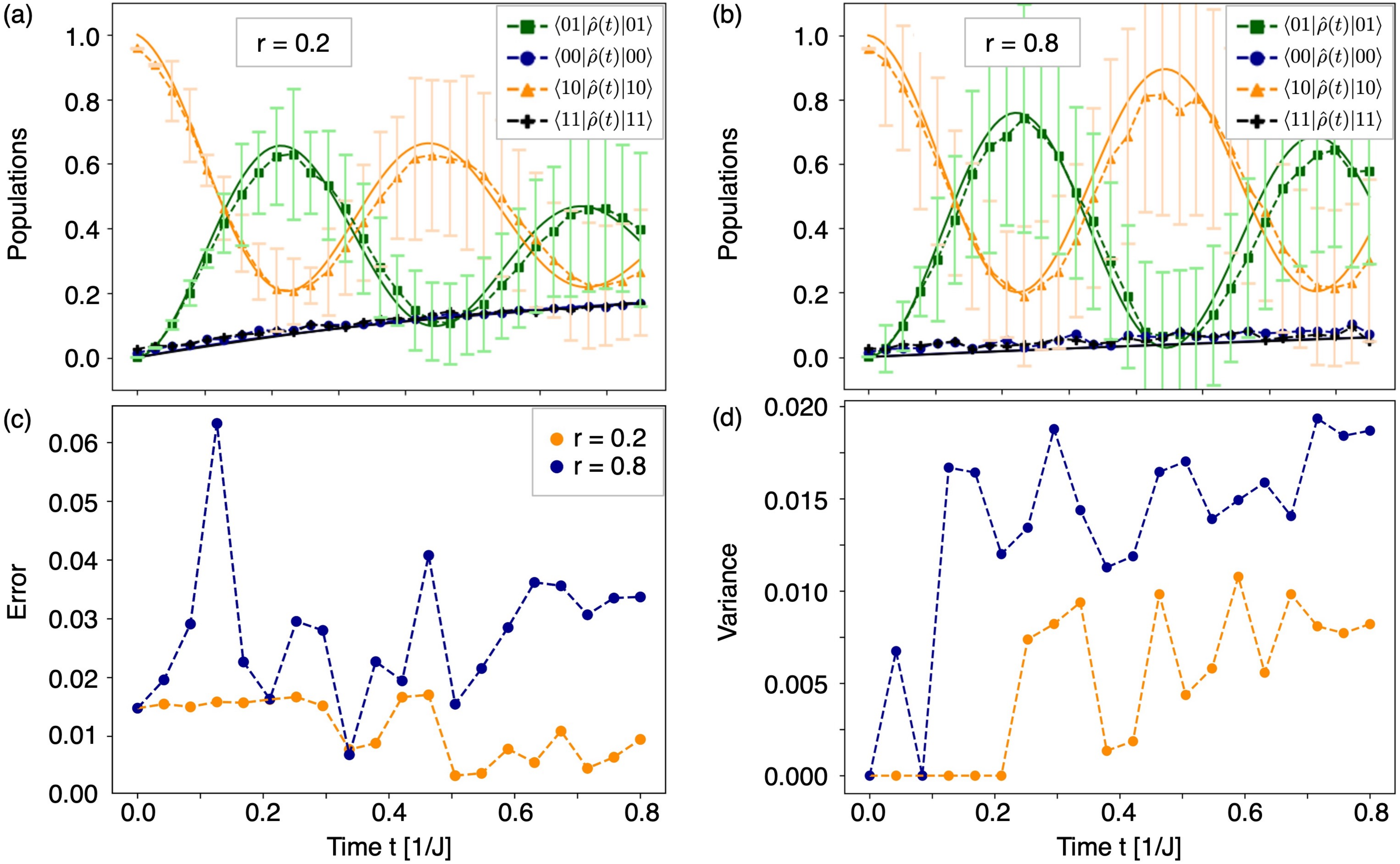
$\mathscr{E}^{-1}(\rho) = \sum_{k=0} q_k$



Implementation

$$\epsilon'_{k>0} = r\epsilon_k$$

$$C = \sum_{k=0} |q_k| = 1 + 2r \sum_{k=1} \epsilon_k$$



Partial Probabilistic Error Cancelation

Mitigation Cost

$$\mathcal{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathcal{P}_k$$

$$p_k = \epsilon'_k / C$$

$$C = \sum_{k=0} |q_k|$$

Mitigation cost

Partial Probabilistic Error Cancelation

Mitigation Cost

$$\mathcal{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathcal{P}_k$$

$$p_k = \epsilon'_k / C$$

$$\Delta O_M \propto M^{-1} \Rightarrow \Delta O_M^{(\text{PEC})} \propto C_{\text{tot}}^2 M^{-1}$$

$$C = \sum_{k=0} |q_k|$$ → Mitigation cost → Variance of the measured observable is increased

Partial Probabilistic Error Cancelation

Mitigation Cost

$$\mathcal{E}^{-1} = C \sum_{k=0} p_k \text{sign}(q_k) \mathcal{P}_k$$

$$p_k = \epsilon'_k / C$$

$$\Delta O_M \propto M^{-1} \Rightarrow \Delta O_M^{(\text{PEC})} \propto C_{\text{tot}}^2 M^{-1}$$

$$C = \sum_{k=0} |q_k|$$ → Mitigation cost → Variance of the measured observable is increased

For a Trotter-type circuit:

$$C_{tot}^2 \sim e^{\lambda n D \epsilon_r}$$

$$\epsilon_r = \sum_{k=1} \epsilon'_k = \sum_{k=1} r_k \epsilon_k$$

Decoherence rate control scheme

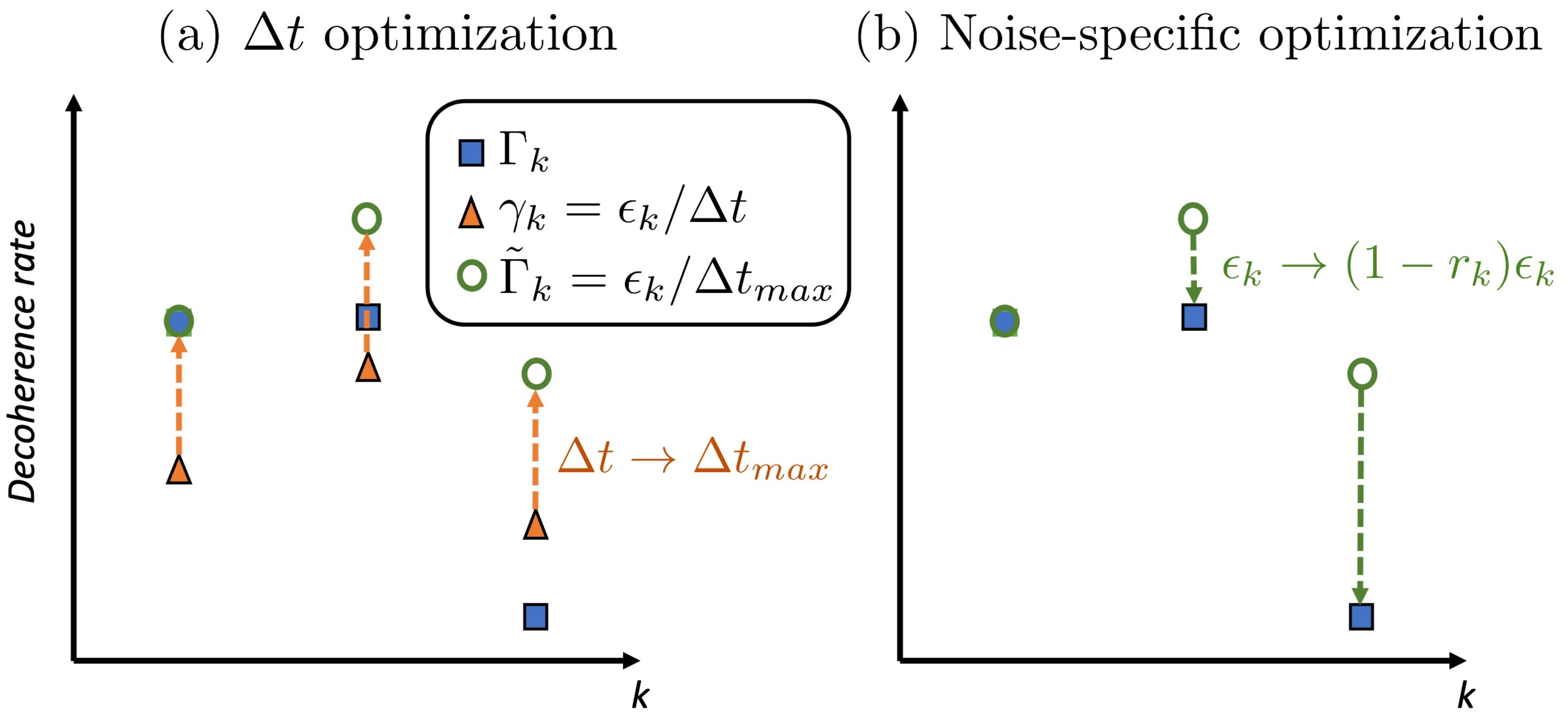
How can we control γ_k in the digital simulation without changing the hardware components?

Target decoherence rates Γ_k : $\mathscr{D}^{\text{(controlled)}}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \Gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t) \right)$

Decoherence rate control scheme

How can we control γ_k in the digital simulation without changing the hardware components?

Target decoherence rates Γ_k : $\mathscr{D}^{(\text{controlled})}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \Gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t)\right)$

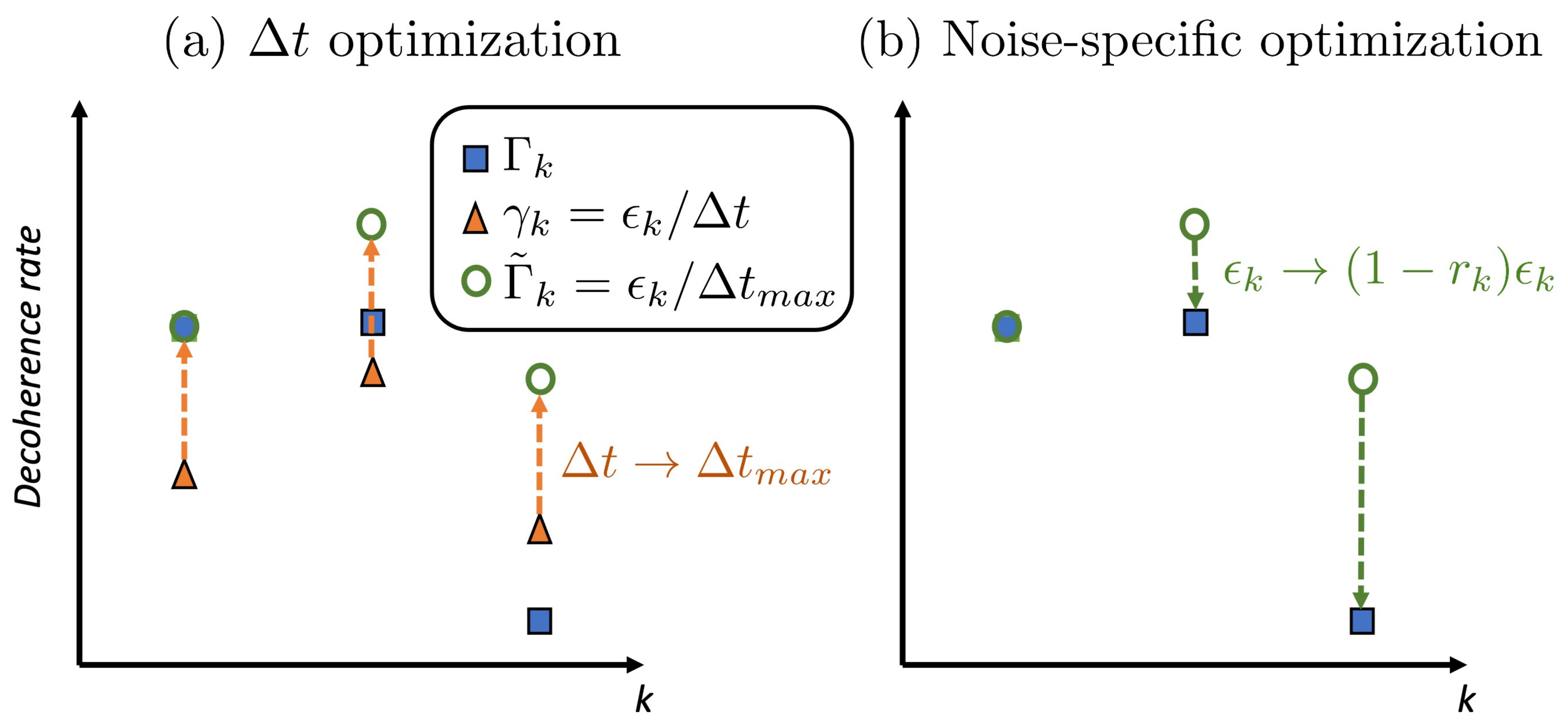


Decoherence rate control scheme

How can we control γ_k in the digital simulation without changing the hardware components?

Target decoherence rates Γ_k :

$$\mathcal{D}^{(\text{controlled})}_{\text{stochastic}}[\hat{\rho}(t)] = \sum_{k=0}^{4^K-1} \Gamma_k \left(\hat{P}_k \hat{\rho}(t) \hat{P}_k - \hat{\rho}(t) \right)$$



$$\Gamma_k = \epsilon_k (1 - r_k) / \Delta t_{max}$$

Limitations of the technique

Target decoherence rates Γ_k : $\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$

$$C_{\text{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\text{max}}^{(\text{NISQ})}$$

Limitations of the technique

Target decoherence rates Γ_k : $\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$

$$C_{\text{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\max}^{(\text{NISQ})}$$

$$\epsilon \lesssim \frac{\Gamma t}{D} + \frac{\ln(M_{\max}^{(\text{NISQ})})}{2\lambda n D^2}$$

$\epsilon = \sum_{k=1} \epsilon_k$ $\Gamma = \sum_{k=1} \Gamma_k$

Limitations of the technique

Target decoherence rates Γ_k : $$\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$$

$$C_{\text{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\max}^{(\text{NISQ})}$$

$$\epsilon \lesssim \frac{\Gamma t}{D} + \frac{\ln(M_{\max}^{(\text{NISQ})})}{2\lambda n D^2}$$

How does D scale?

$\epsilon = \sum_{k=1} \epsilon_k$ $\quad$ $\Gamma = \sum_{k=1} \Gamma_k$

Limitations of the technique

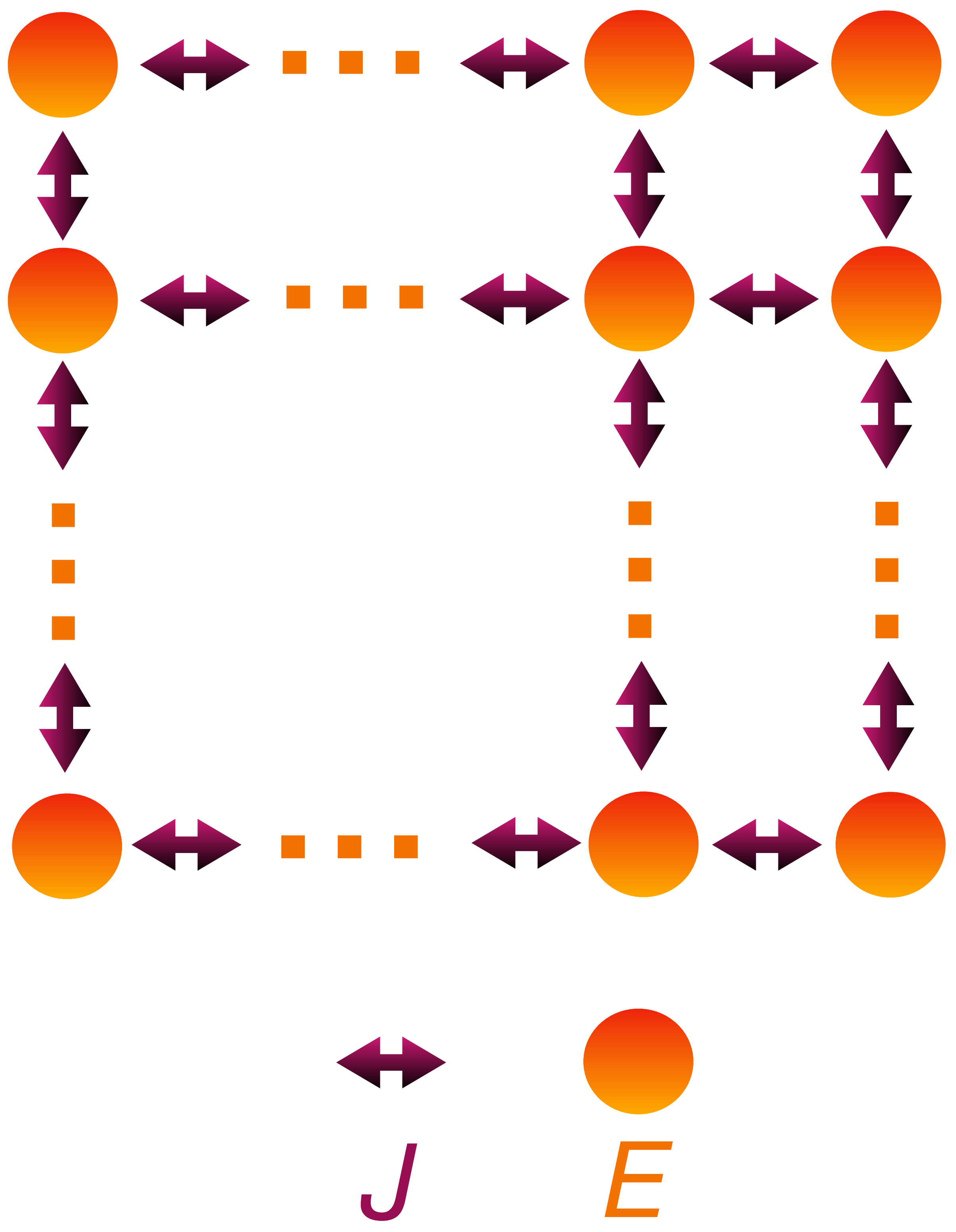
Target decoherence rates Γ_k : $\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$

$$C_{\text{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\max}^{(\text{NISQ})}$$

$$\epsilon \lesssim \frac{\Gamma t}{D} + \frac{\ln(M_{\max}^{(\text{NISQ})})}{2\lambda n D^2}$$

$\epsilon = \sum_{k=1} \epsilon_k \qquad \Gamma = \sum_{k=1} \Gamma_k$

How does D scale?



$$\hat{H} = -\sum_{m=1}^{n} \frac{E}{2}\hat{Z}_m + \sum_{\langle m,m' \rangle} \frac{J}{2}(\hat{X}_m\hat{X}_{m'} + \hat{Y}_m\hat{Y}_{m'})$$

Limitations of the technique

Target decoherence rates Γ_k :

$$\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$$

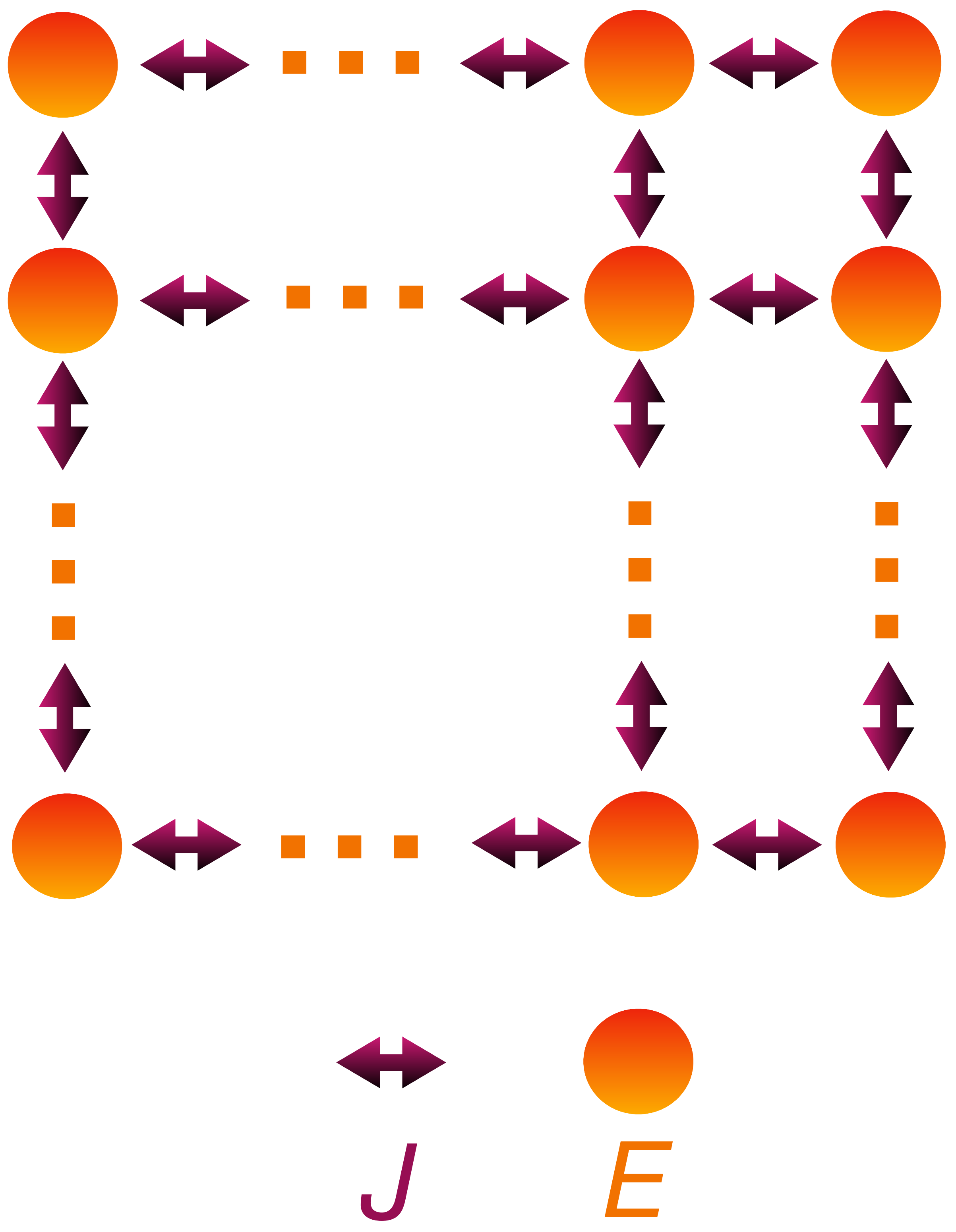
$$C_{\mathrm{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\max}^{(\mathrm{NISQ})}$$

$$\epsilon \lesssim \frac{\Gamma t}{D} + \frac{\ln(M_{\max}^{(\mathrm{NISQ})})}{2\lambda n D^2}$$

$$\epsilon = \sum_{k=1} \epsilon_k \qquad \Gamma = \sum_{k=1} \Gamma_k$$

Childs, Andrew M., et al. Physical Review X 11.1 (2021): 011020.

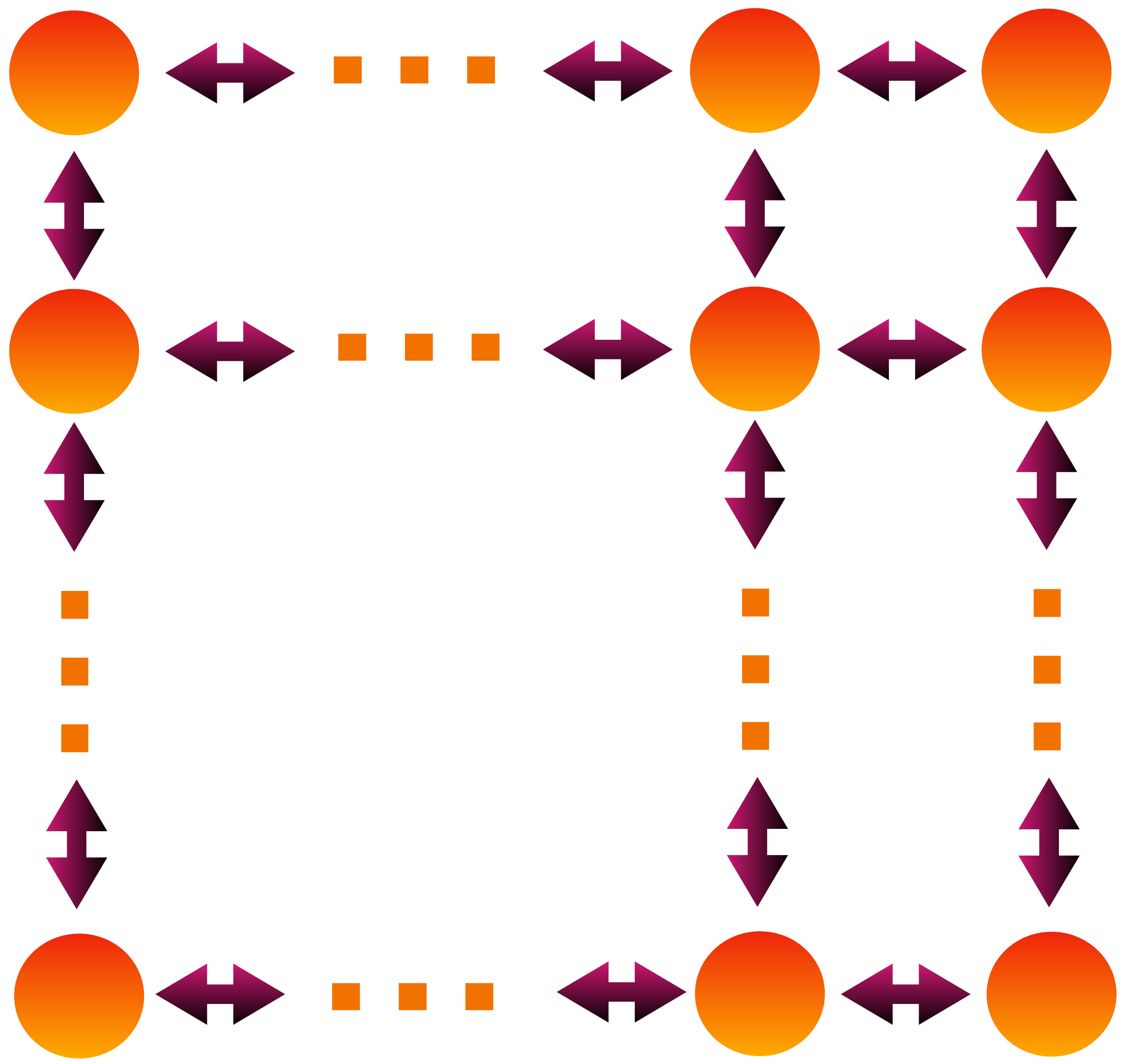
$$D = O\left(\frac{n^d J(J+E)t^2}{\varepsilon_{\mathrm{Trot}}}\right)$$



$$\hat{H} = -\sum_{m=1}^{n} \frac{E}{2}\hat{Z}_m + \sum_{\langle m,m'\rangle} \frac{J}{2}(\hat{X}_m\hat{X}_{m'} + \hat{Y}_m\hat{Y}_{m'})$$

Limitations of the technique

Target decoherence rates Γ_k : $\Gamma_k = \epsilon_k(1 - r_k)/\Delta t$



$$C_{\text{tot}}^2 \sim e^{\lambda n D \epsilon_r} \lesssim M_{\max}^{(\text{NISQ})}$$

$$D = O\left(\frac{n^d J(J+E)t^2}{\varepsilon_{\text{Trot}}}\right)$$

$$\epsilon \lesssim \frac{\Gamma t}{D} + \frac{\ln(M_{\max}^{(\text{NISQ})})}{2\lambda n D^2}$$

$$\epsilon \lesssim O\left(\frac{\varepsilon_{\text{Trot}}\Gamma}{n^d J(J+E)t} + \frac{\varepsilon_{\text{Trot}}^2 \ln(M_{\max}^{(\text{NISQ})})}{2\lambda n^{2d+1} J^2 (J+E)^2 t^4}\right)$$

$\epsilon = \sum_{k=1} \epsilon_k$ $\quad$ $\Gamma = \sum_{k=1} \Gamma_k$

Conclusion

Noise-assisted simulation

- Applications: NISQ computers and quantum analog simulations.

Conclusion

Noise-assisted simulation

- Applications: NISQ computers and quantum analog simulations.
- The quantum hardware does not need to be changed in order to tune the noise rate;

Conclusion

Noise-assisted simulation

- Applications: NISQ computers and quantum analog simulations.
- The quantum hardware does not need to be changed in order to tune the noise rate;
- Open quantum systems coupled to Markovian environments via stochastic Pauli noise channels with medium- or large-decoherence rates.

Note: Technique is not restricted only to stochastic Pauli channels

Conclusion

Noise-assisted simulation

- Applications: NISQ computers and quantum analog simulations.
- The quantum hardware does not need to be changed in order to tune the noise rate;
- Open quantum systems coupled to Markovian environments via stochastic Pauli noise channels with medium- or large-decoherence rates.
- Ongoing work: Generalization of this technique to simulate non-perturbative dynamics of open quantum systems.

Thank you for your attention!

José D. Guimarães | 2023