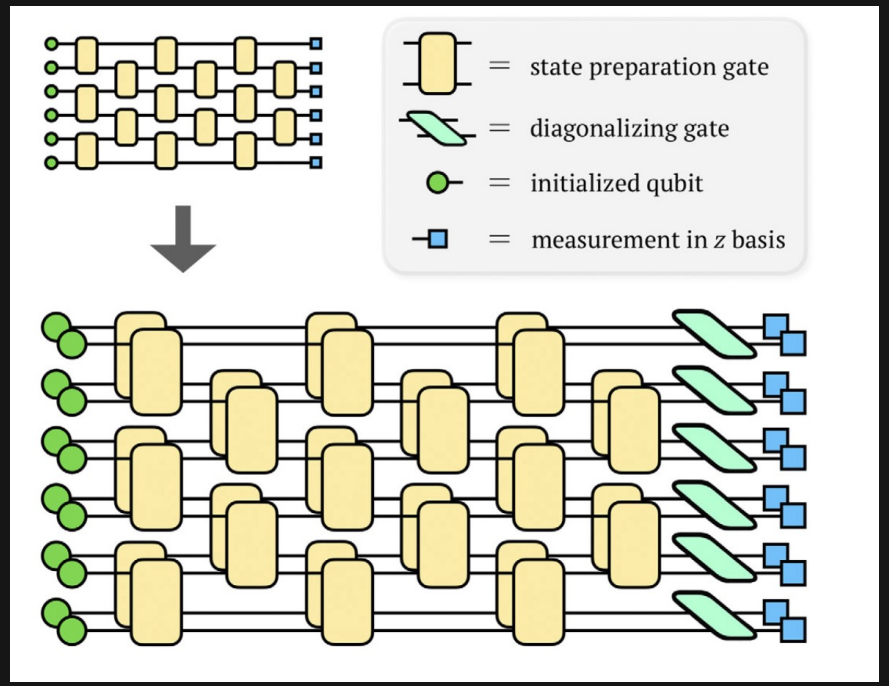
Implement virtual distillation, an error mitigation technique, in Qiskit

1. **Error mitigation** is vital to enhance the performance of NISQ devices.
2. We wish to mitigate **incoherent** noise with virtual distillation.
3. We use a variant, called **dual state verification**, which requires less connectivity.

Virtual Distillation:

Uses multiples copies to supress the non-dominating components exponentially fast.



- requires connectivity between every copies.
- The diagonalising gate can be challenging to implement in general.

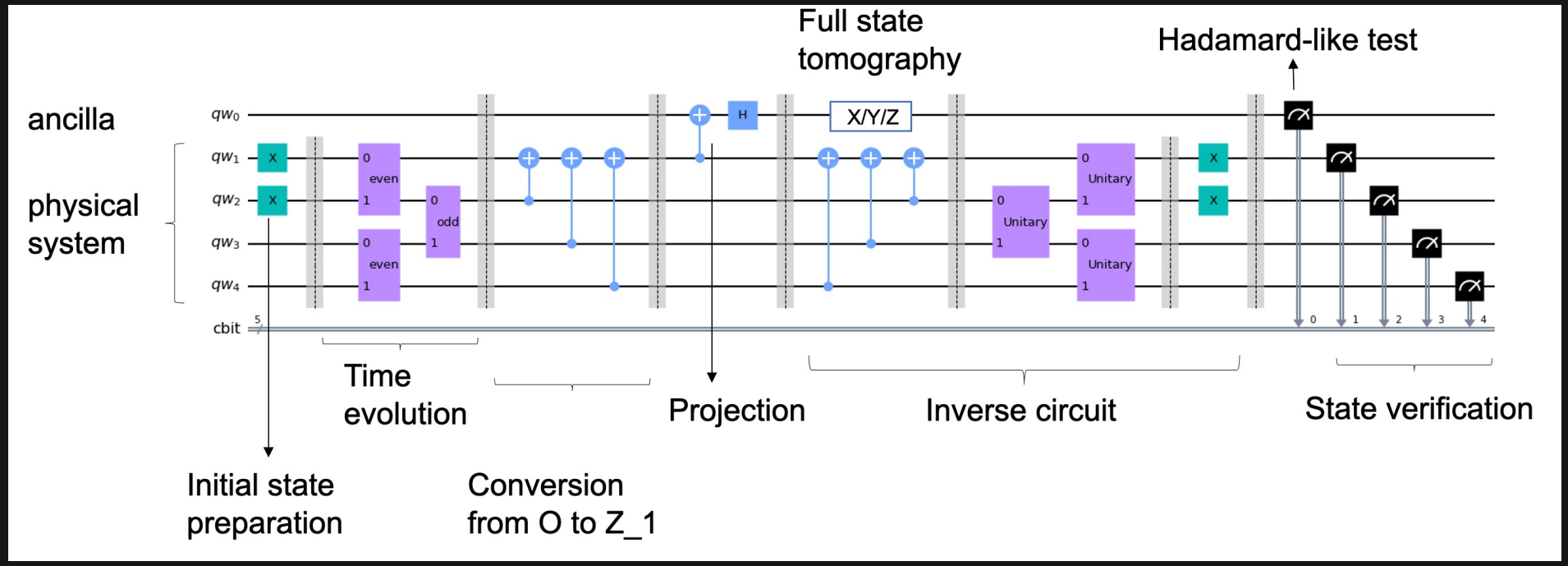
Huggins, et al., Phys. Rev. X 11, 041036 (2021)

Dual State Verification (Echo verification)

Mingxia Huo and Ying Li Phys. Rev. A 105, 022427 (2022)

Thomas E. O'Brien et al., PRX Quantum 2, 020317 (2021)

1. We need to convert the observable via $O = BZ_1B^\dagger$.
2. The expectation value is measured via a non-destructive measurement on an ancilla.
3. In the noise-less regime, the ancilla sould be pure.
4. We perform full tomogrphy to purify the ancilla qubit.

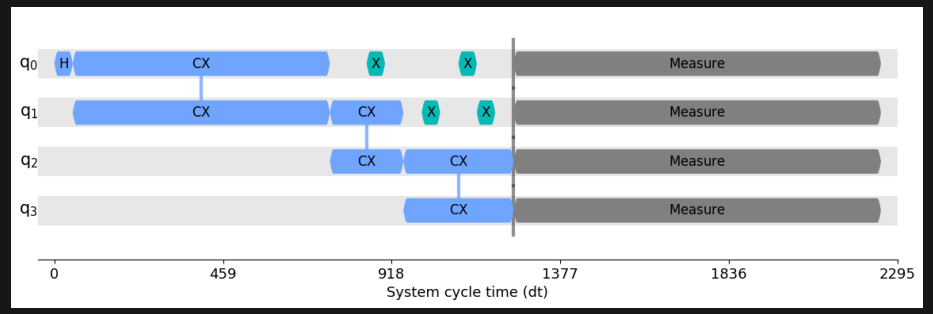


IBM **Quantum**

Improving Virtual Distillation? Turn coherent errors to incoherent ones!

Dynamical Decoupling

Decrease decoherence by taking advantage of **rapid, time-dependent control modulation** when the qubits are idling.



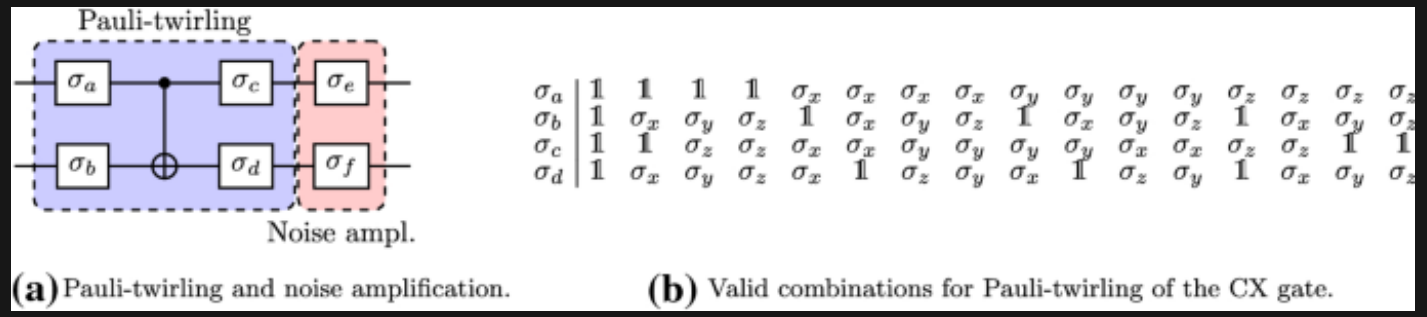
Which pulses sequences should we use?

(XY8)

Pauli Twirling

Turn a noisy operator into a Pauli channel, via gate conjugation.

$$\mathcal{T}_W(\overline{M}) = \frac{1}{|W|}\sum_{w\in W}\overline{wMw^\dagger}\,.$$



Fuchs, et al., *Eur. Phys. J. Plus* **135**, 353 (2020)

Software demonstration

https://github.com/orielkiss/qiskit-research/tree/DSV_tutorial/qiskit_research/DSV

Inputs:

- construct_circuit(params): prepare your circuits
- measured_operator: pauli_string to be measured
- psi: initial state

Outputs:

- Expectation value at each Trotter step for the raw, DSV and DSV purified circuits.
- Purity of the ancilla qubit.

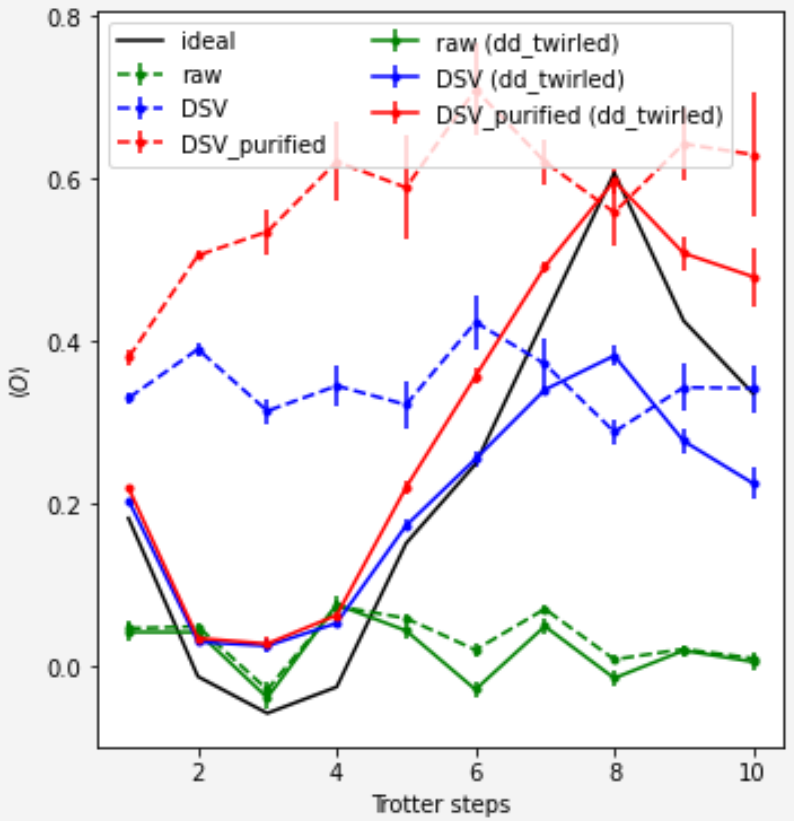
```
DSV = Dual_state_verification()
DSV._step = step
qc_list = []

for i in range(1,step+1):
    qc_origin = construct_circuit(N, J, delta, i, psi)

    for anc_op in ['X', 'Y', 'Z']:
        qc_list.append(DSV.prepare_vd_circuit_arbitrary(qc_origin, measured_operator, False, anc_op))

qc_transpiled = DSV.transpile(qc_list, backend = backend, initial_layout = initial_layout,
                              num_twirls =0, dynamical_decoupling= None,
                              optimization_level = 3, seed = 17245)

job = qiskit.execute(qc_transpiled, backend= backend_noisy, shots = shots)
counts = job.result().get_counts()

counts = DSV.post_process(counts)
expectation,purity = DSV.compute_expectation_value(counts, measured_operator)
```

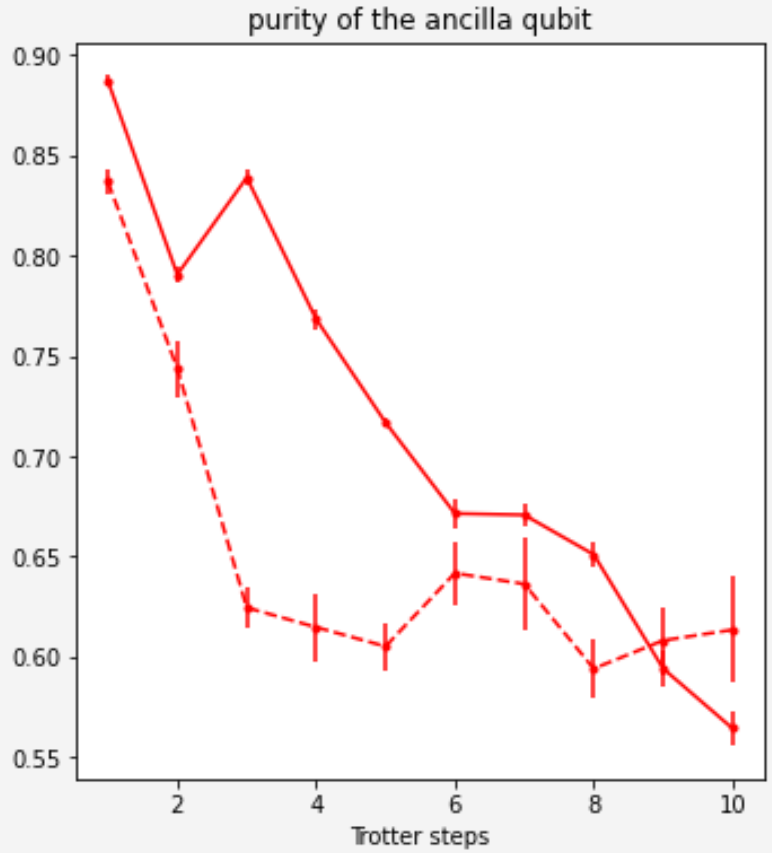
Experiment (IBMQ_mumbai)

$$H = J \sum_{j=1}^{9} X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1},$$

- VD alone is not sufficient (needs purification).
- Twirling and DD improves the purity.
- Twirling and DD have little effect on the raw data.

O = ZZIIIIII





Some more

O = ZIIIIIII

- VD alone is not sufficient (needs purification).
- XY8 DD helps.
- Twirling improves the purity.
- XY8 DD + Twirling + purified VD is the best.