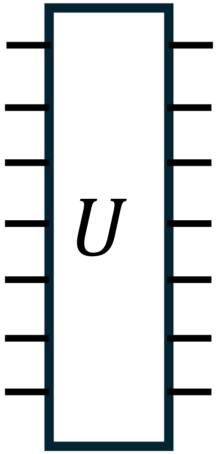
Learning shallow quantum circuits and quantum states prepared by shallow circuits in polynomial time

Yunchao Liu (UC Berkeley)

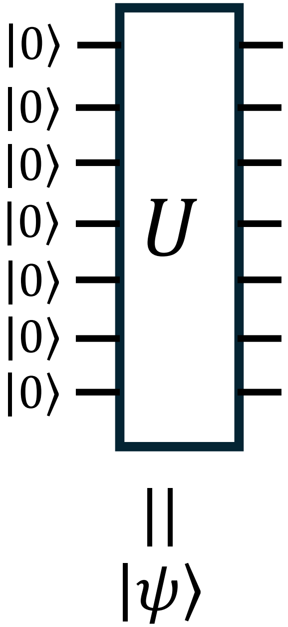
Based on arxiv 2401.10095, and upcoming work with Zeph Landau

Two fundamental problems

- Given access to an unknown constant depth quantum circuit U, learn a constant depth circuit that is close to U



- Given copies of an unknown quantum state $|\psi\rangle = U|0^n\rangle$ that is prepared by an unknown constant depth circuit U, learn a constant depth circuit that prepares $|\psi\rangle$



- **This talk: polynomial time algorithms for both problems**
  - quasi-polynomial time, when depth of U is $\mathrm{polylog}(n)$

Quantum algorithms in NISQ

- NISQ computation can be modeled as shallow quantum circuits
  - Can generate probability distributions that are classically hard
- Key idea behind NISQ algorithms: try to discover a shallow circuit as a solution to an interesting problem (assuming the circuit exists)
  - Can be formulated as a learning problem
- Main challenge: how to develop efficient learning algorithms?
  - This talk: two new learning algorithms that provably work in simple settings
  - Primitives for new NISQ algorithms?

Key challenge: efficient reconstruction

- Step 1: Learn local observables
  - Easy to do (using e.g. classical shadows)
  - Sufficient information

- **Step 2: efficiently reconstruct a quantum circuit from learned local observables**
  - This is a highly non-trivial problem
  - **Goal of this talk: demonstrate new and simple techniques to do this**

Learning shallow quantum circuits

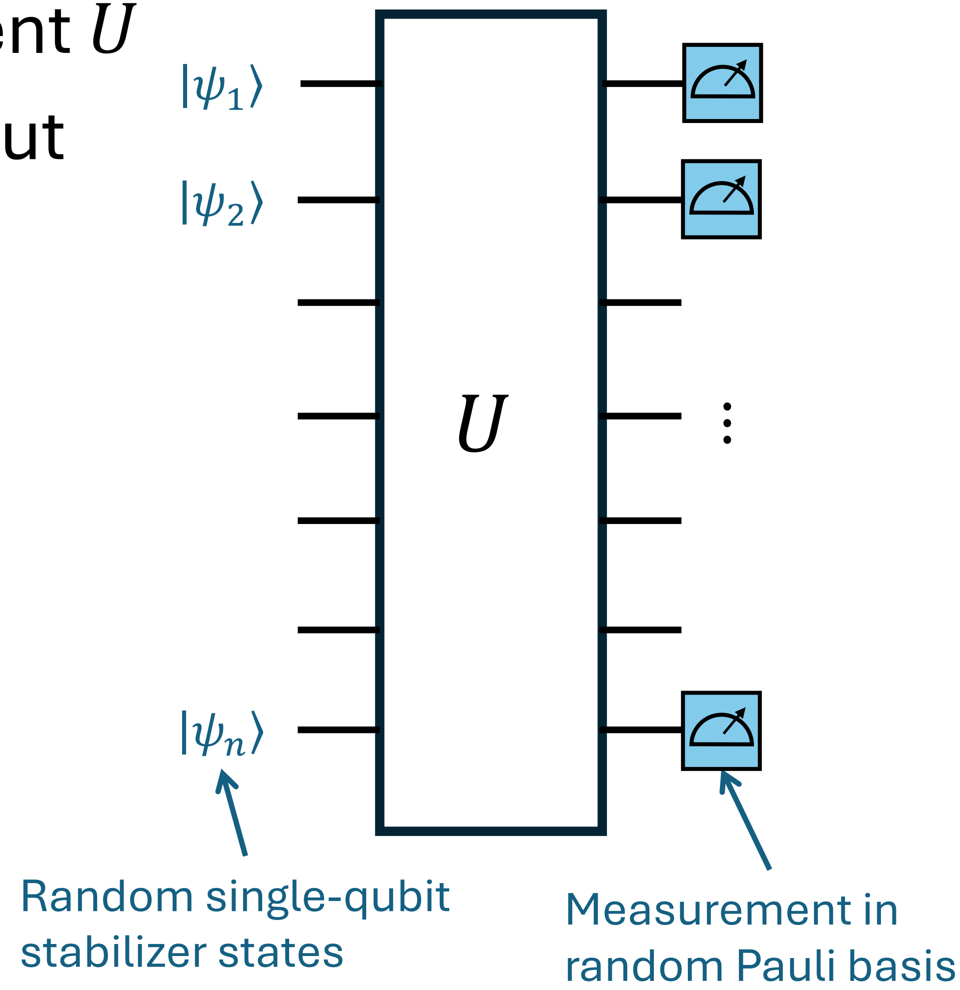
Based on arxiv 2401.10095 (QIP 2024, STOC 2024)

with Hsin-Yuan Huang, Michael Broughton, Isaac Kim, Anurag Anshu, Zeph Landau, Jarrod R. McClean

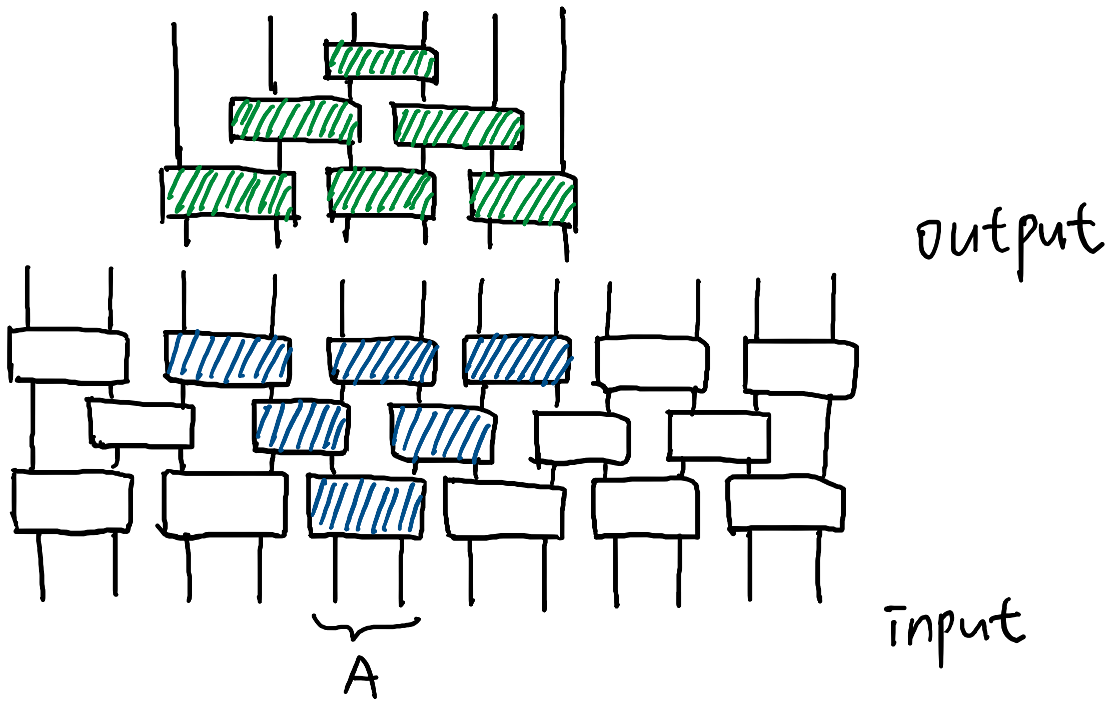
Learning shallow quantum circuits

- Want to learn a shallow circuit to implement U
- Only need single-qubit random input/output samples

- **Theorem.** Polynomial time algorithm for learning shallow quantum circuits from random input/output samples



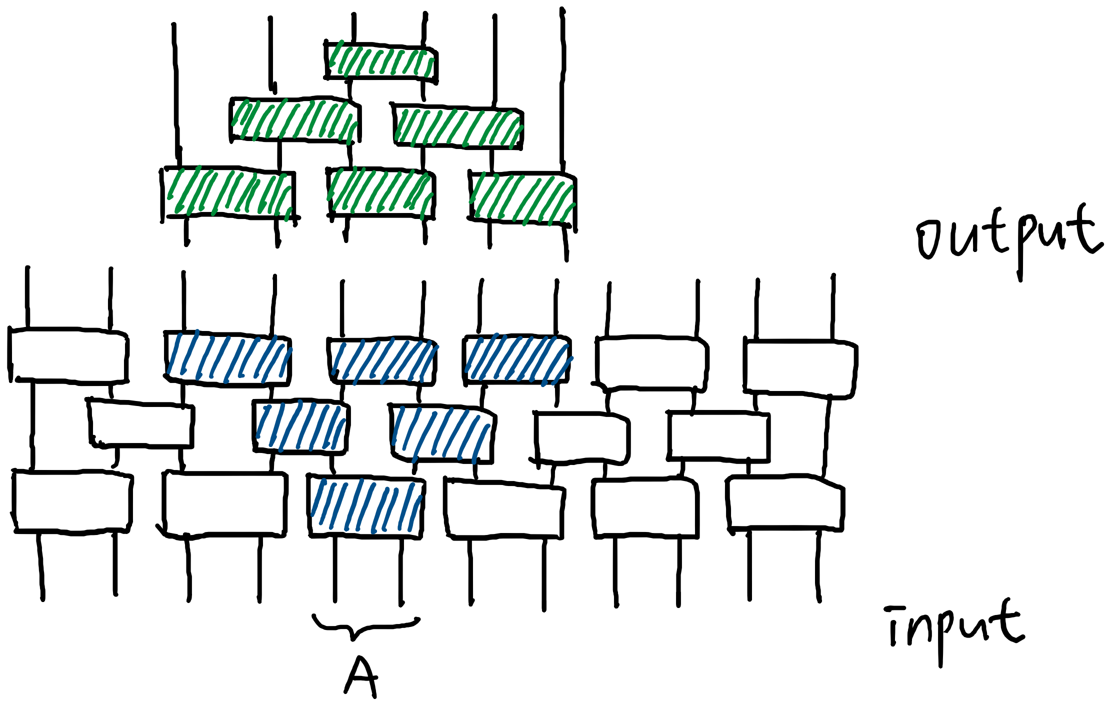
Lightcone in shallow circuits



Basic idea: guess a **small circuit** to disentangle the region A

- Consider a small region A (on a lattice), each input qubit in A only affects the output qubits in the lightcone of A
- If we can **undo** the blue gates ➔ U acts as identity on A

Basic idea: local inversion

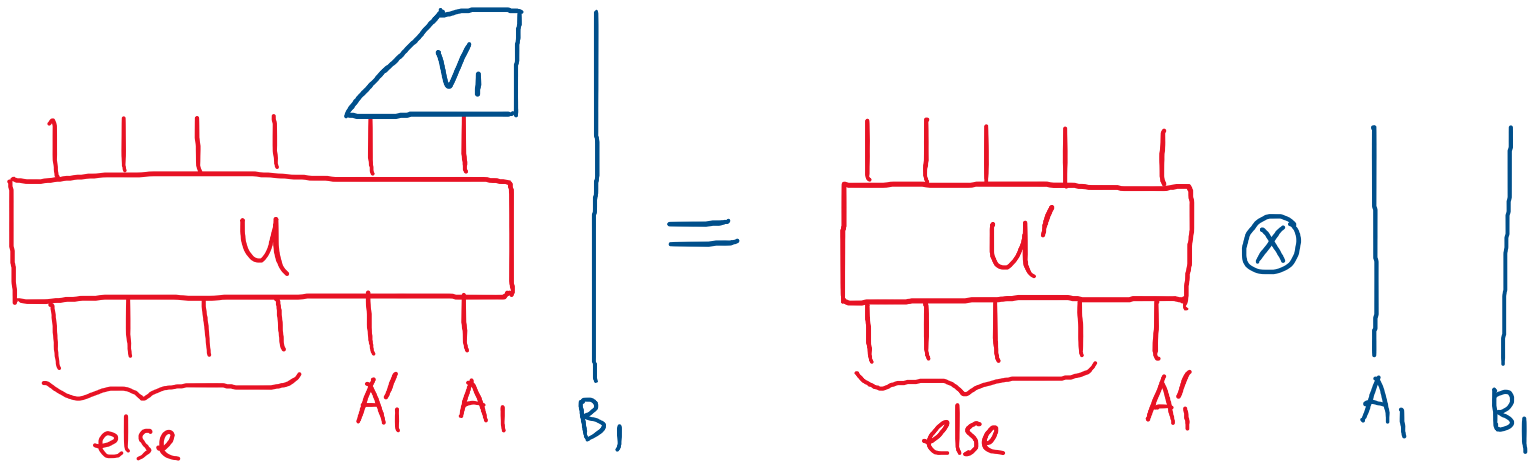


Local inversion: guess a **small circuit** to disentangle the region A

How? Enumerate **small circuits** and test if A is disentangled

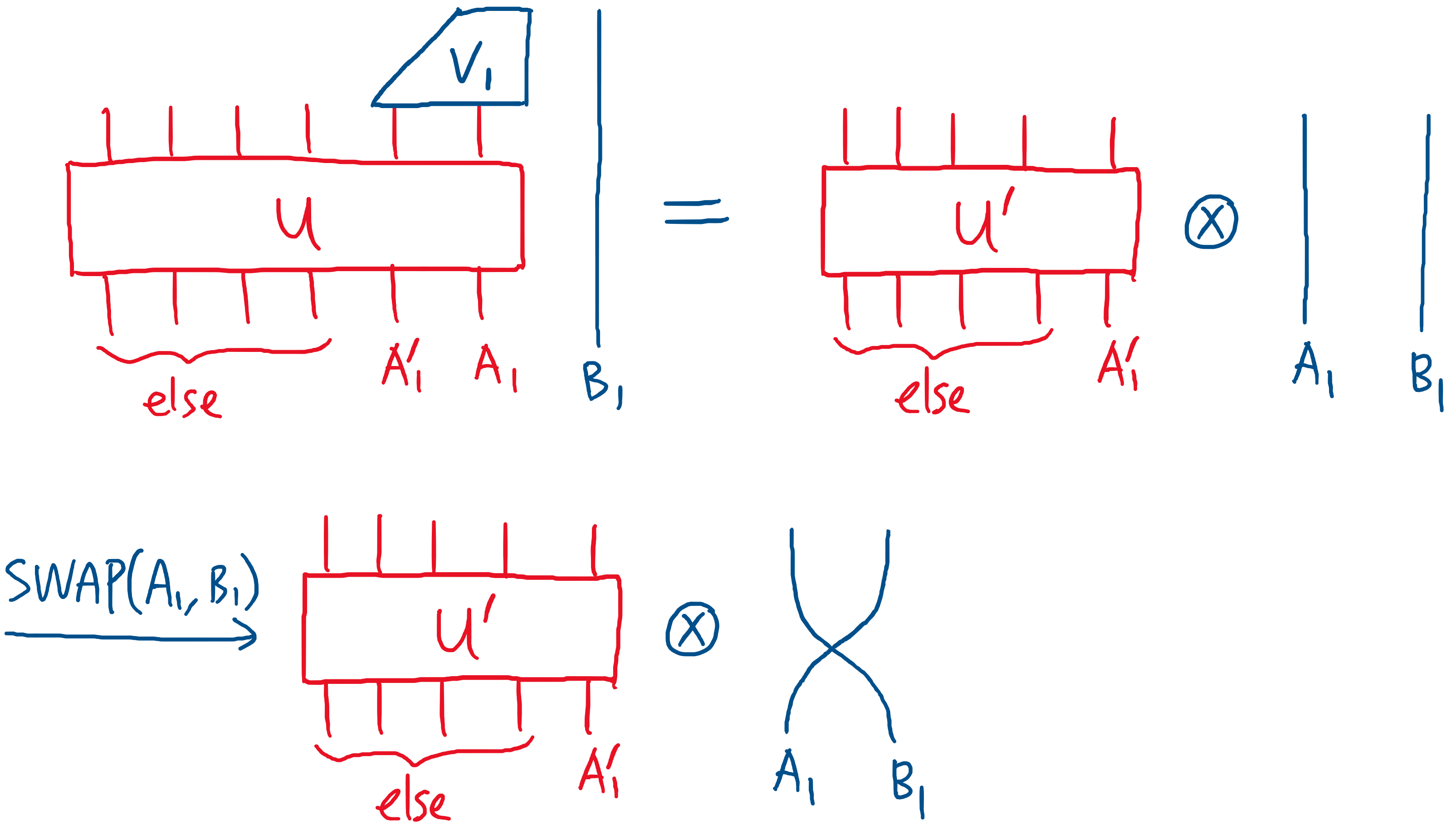
Key issue: the small circuit we apply could be different from the actual lightcone; in this case it creates a mess on remaining qubits!

Key idea: disentangle, swap, *undo*



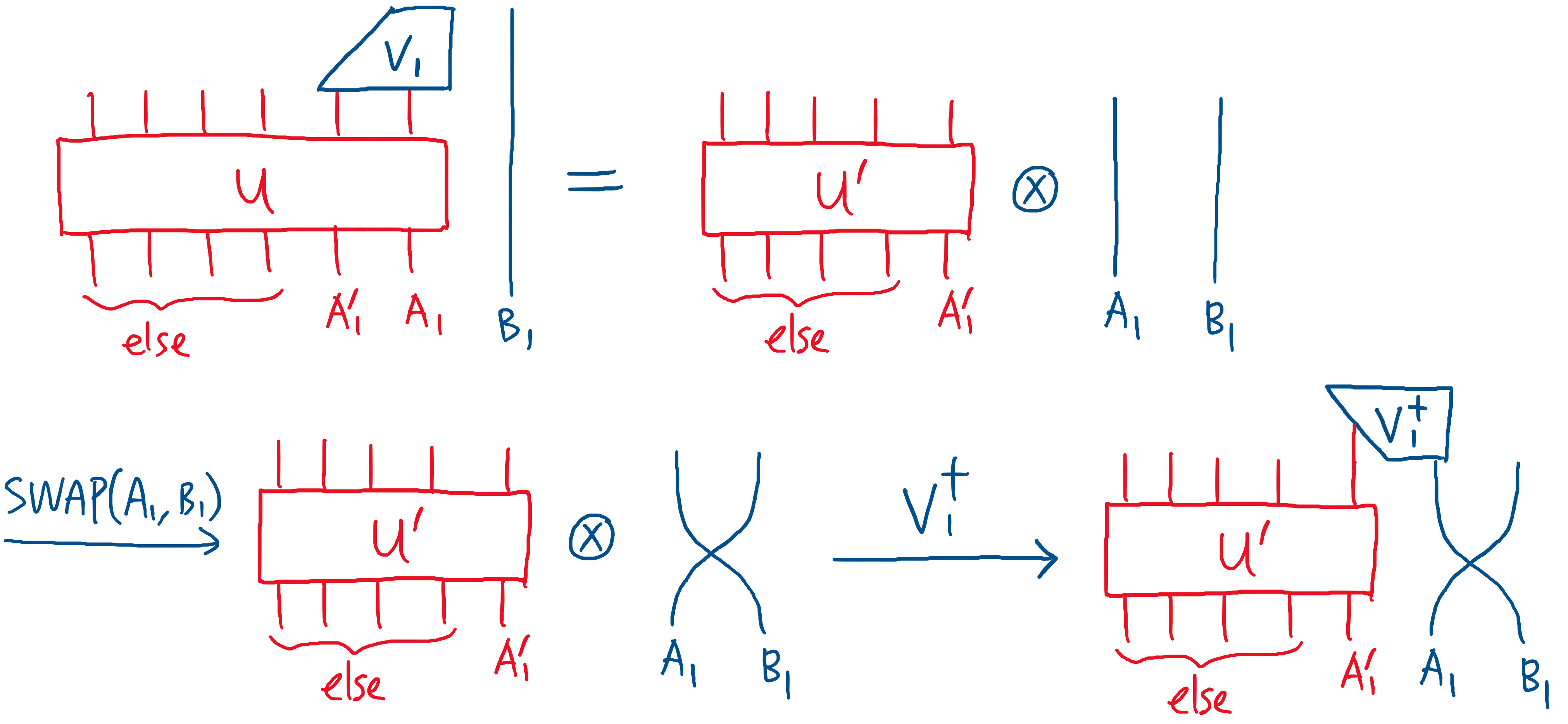
Key idea: introduce a new ancilla qubit, swap with A_1, then *undo* V_1

Key idea: disentangle, swap, *undo*



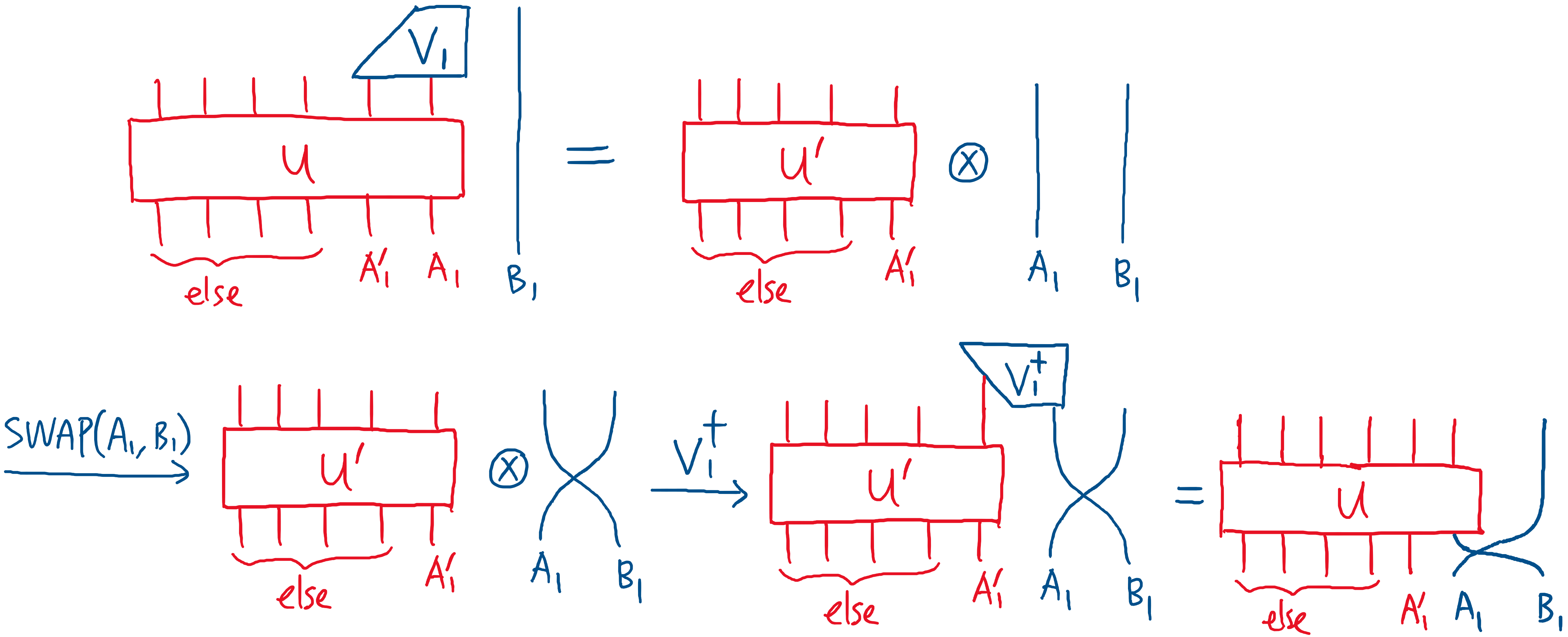
Key idea: introduce a new ancilla qubit, swap with A_1, then *undo* V_1

Key idea: disentangle, swap, *undo*



Key idea: introduce a new ancilla qubit, swap with A_1, then *undo* V_1

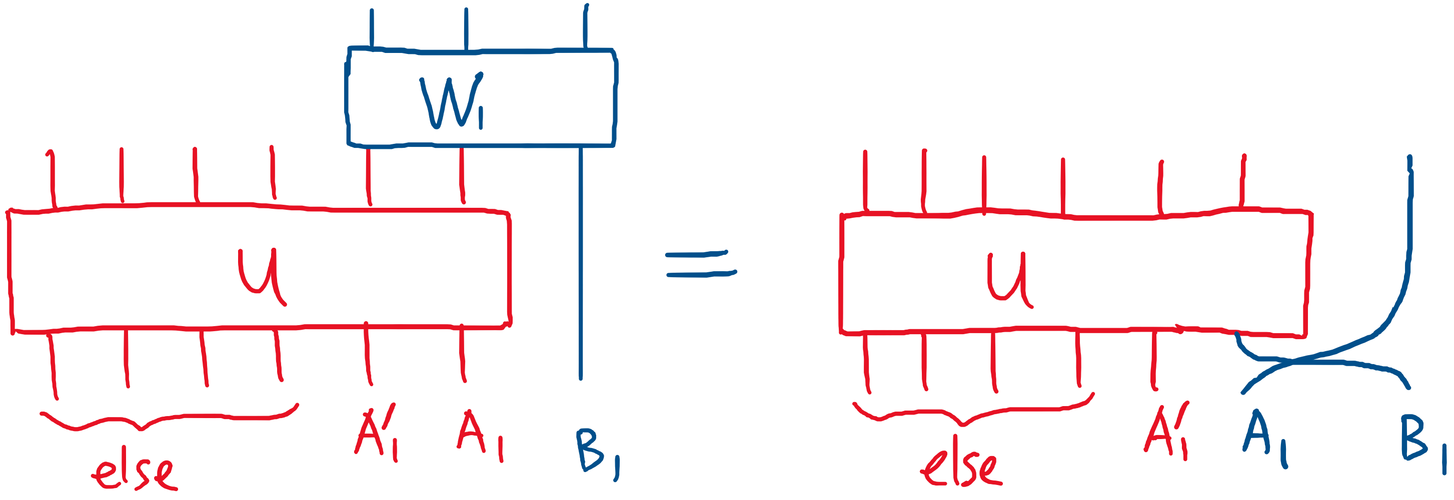
Key idea: disentangle, swap, *undo*



Key idea: introduce a new ancilla qubit, swap with A_1, then *undo* V_1

Key idea: disentangle, swap, *undo*

- We have learned a unitary W_1 such that

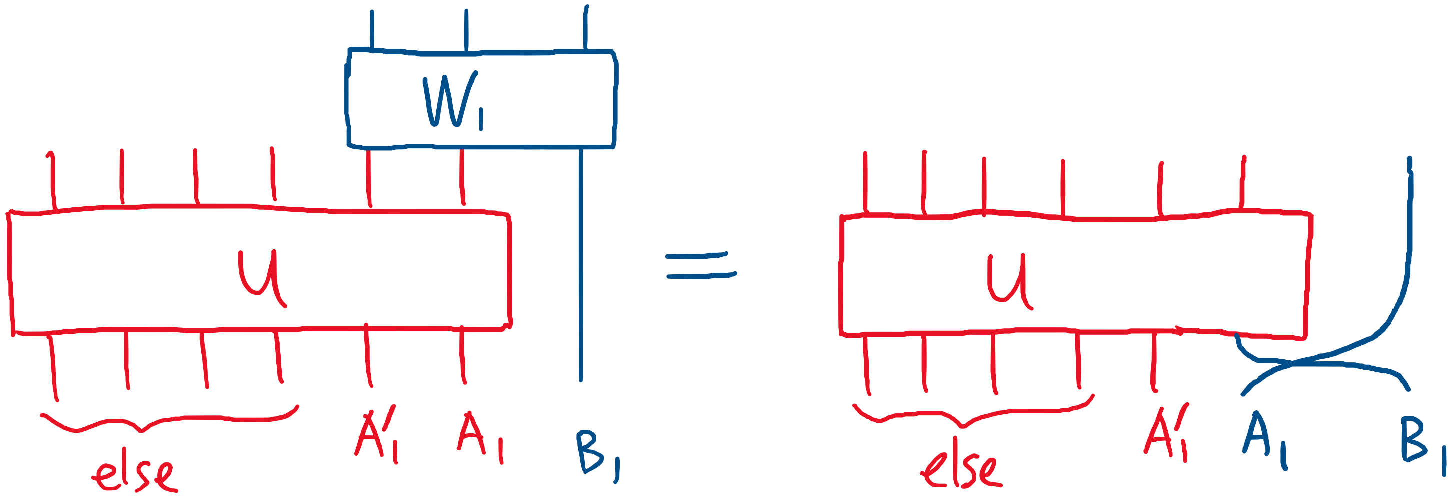


We have learned a circuit acting on top of U that achieves the effect of swapping an input wire

Key observation: the system is not disturbed; can repeat this for every qubit

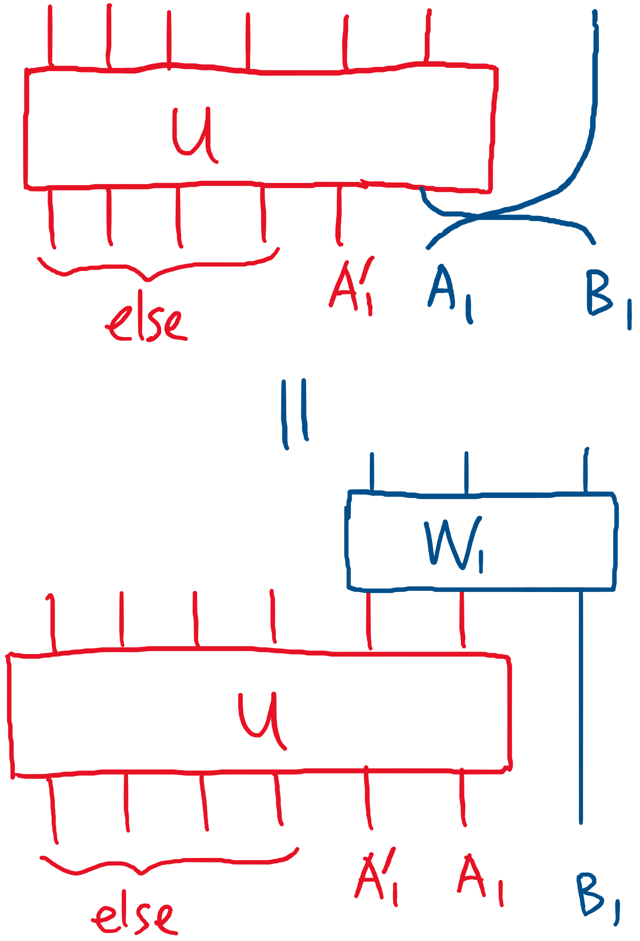
Reconstructing the circuit

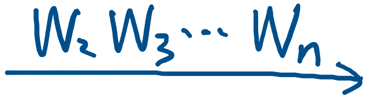
- We have learned a unitary W_1 such that



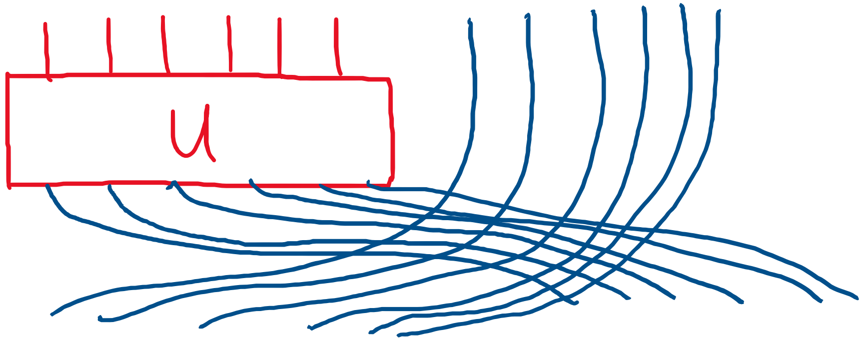
Claim: using this idea, can learn the description of a $2n$-qubit circuit W that satisfies $W = U^\dagger \otimes U$

Reconstructing the circuit

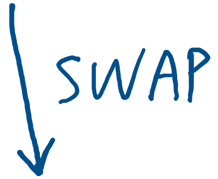


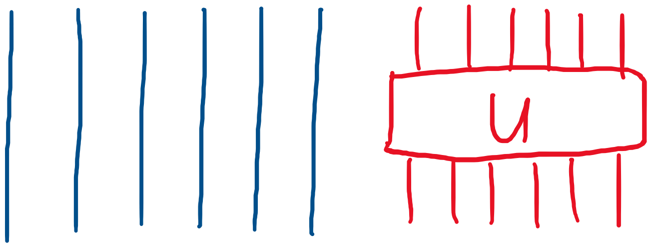


Do the same
for every qubit



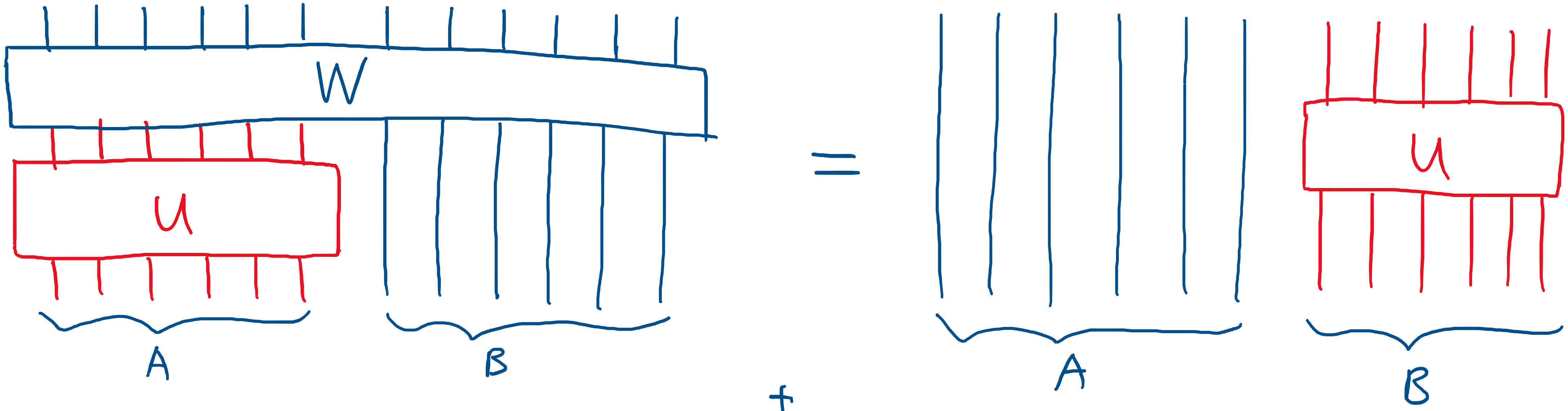
SWAP entire system
with ancilla





Reconstructing the circuit

- We have learned a $2n$-qubit circuit W that satisfies

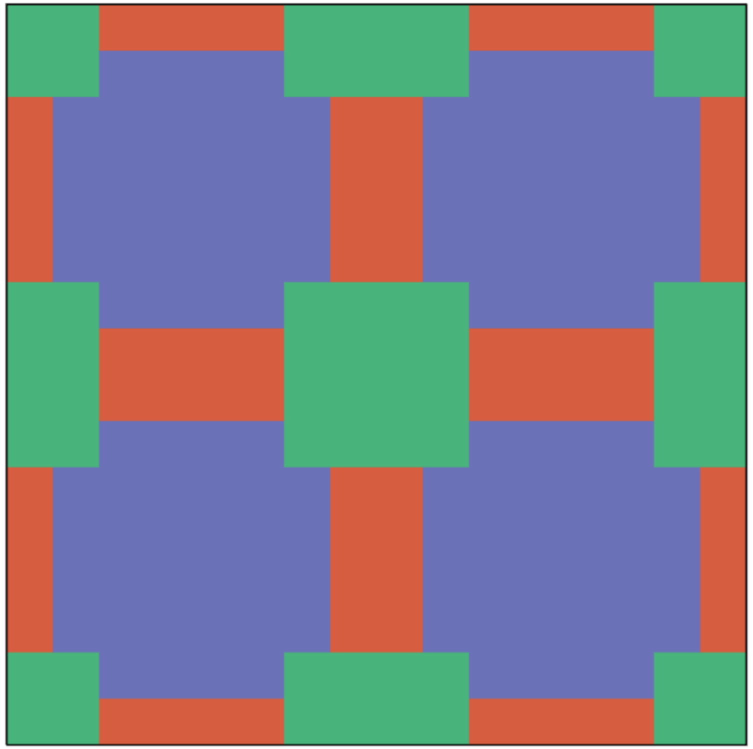


$$\Rightarrow W = U_A^\dagger \otimes U_B$$

disentangle a local region without disturbing the system

Reconstructing the circuit in low depth

- 2D example: reconstruct the circuit in 3 layers

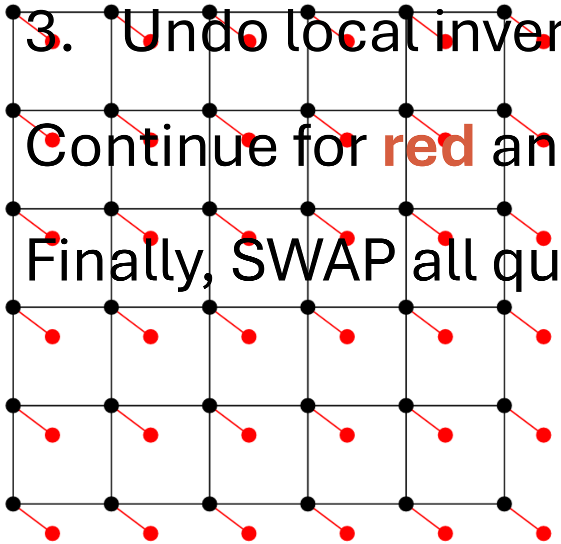


Do this in parallel for all **green** regions:

1. Apply local inversion operator
2. SWAP the entire region with ancilla wires
3. Undo local inversion operator

Continue for **red** and **blue**

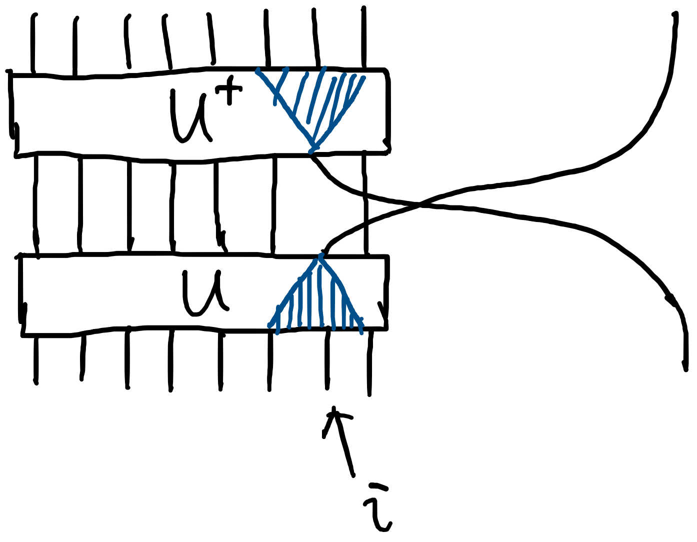
Finally, SWAP all qubits with ancilla



Each qubit is connected to an **ancilla** qubit

Further simplifying the argument

- Key observation: steps 1-3 can be merged into one step, which is equivalent to implementing $U^\dagger S_i U$ (S_i: SWAP i-th qubit and i-th ancilla)
  - Sanity check: $U^\dagger S_i U$ is a local operator



Further simplifying the argument

- Key observation: steps 1-3 can be merged into one step, which is equivalent to implementing $U^\dagger S_i U$ (S_i: SWAP i-th qubit and i-th ancilla)
- Simpler algorithm: can directly learn these local operators $U^\dagger S_i U$ and then combine them into a circuit that implements $U \otimes U^\dagger$

One line proof: an identity for any unitary U

$$U \otimes U^\dagger = \left(\prod_{i=1}^{n} S_i\right) \cdot \prod_{i=1}^{n} (U^\dagger S_i U)$$

- Extension: can learn any unitary that maps a local operator to a local operator (quantum cellular automata)

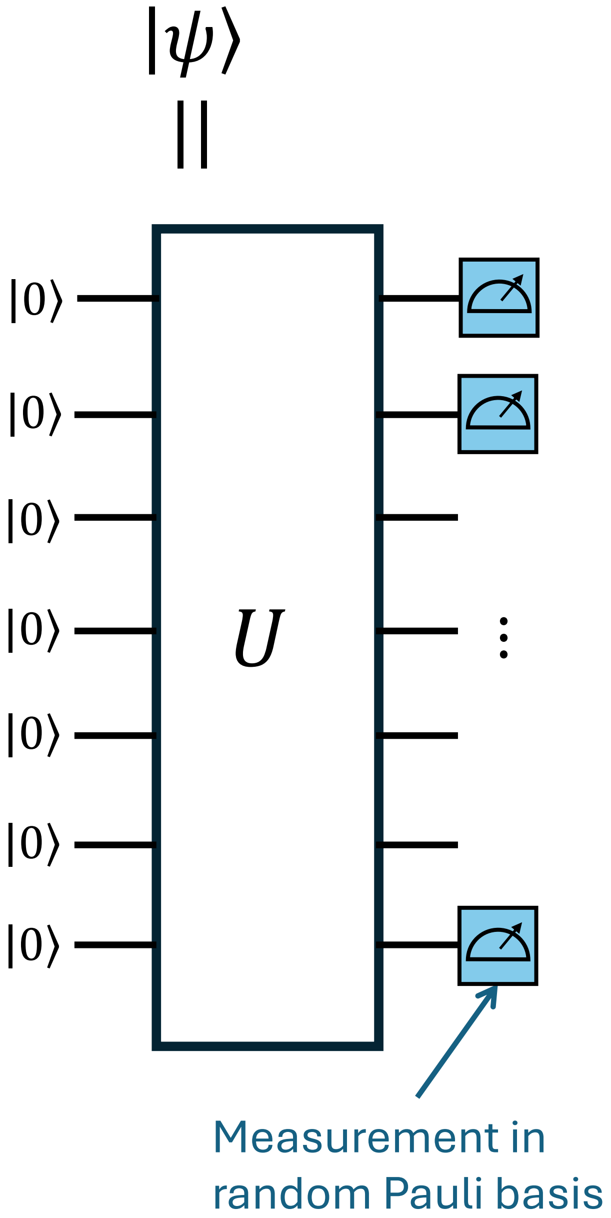
Learning quantum states prepared by shallow circuits

Based on upcoming work with Zeph Landau

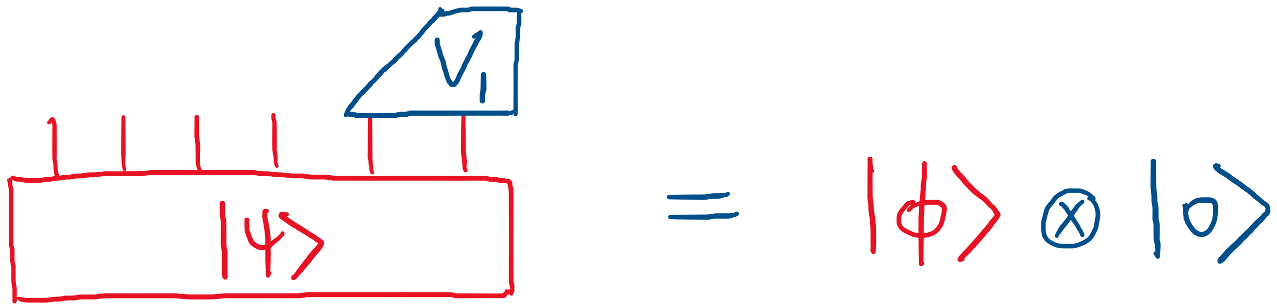
See related upcoming work of Hyun-Soo Kim, Isaac Kim, and Daniel Ranard for a different approach

Learning quantum states

- Want to learn a shallow circuit to prepare $|\psi\rangle$
- Only need single-qubit random measurement samples
- **Theorem.** Polynomial time algorithm for learning quantum states prepared by shallow circuits from random measurement samples



Basic idea: local inversion

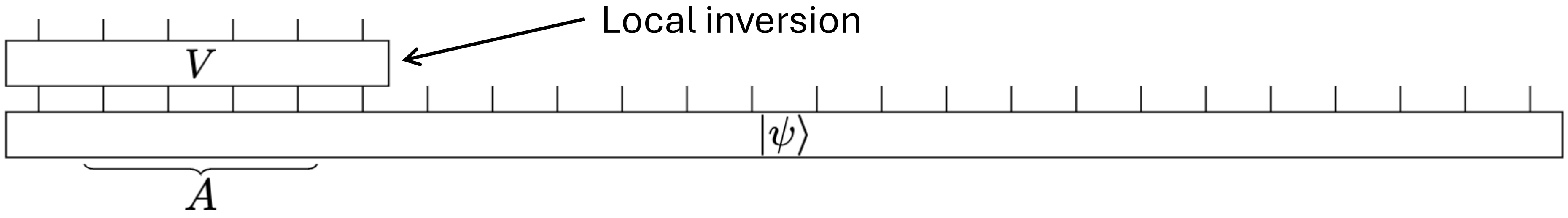


Local inversion: find a small circuit to invert a small region to $|0\rangle$

What to do next?
Similar issues: $|\phi\rangle$ could be a much more complicated state

Key idea 1: local inversion, *undo*

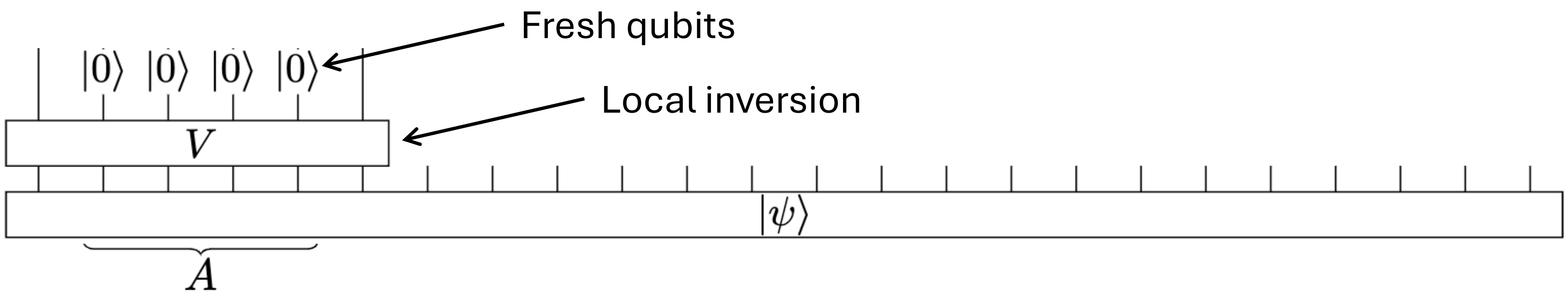
Step 1: Apply a local inversion V to invert the region A, get $|\phi\rangle \otimes |0\rangle_A$



Key idea 1: local inversion, *undo*

Step 1: Apply a local inversion V to invert the region A, get $|\phi\rangle \otimes |0\rangle_A$

Step 2 (not doing anything): replace A with fresh qubits in state $|0\rangle_A$

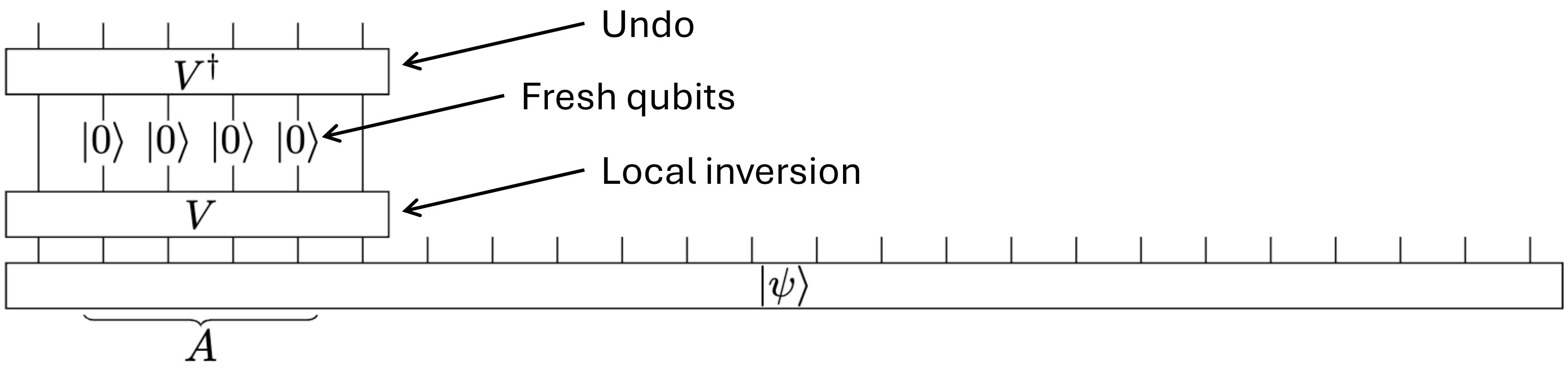


Key idea 1: local inversion, *undo*

Step 1: Apply a local inversion V to invert the region A, get $|\phi\rangle \otimes |0\rangle_A$

Step 2 (not doing anything): replace A with fresh qubits in state $|0\rangle_A$

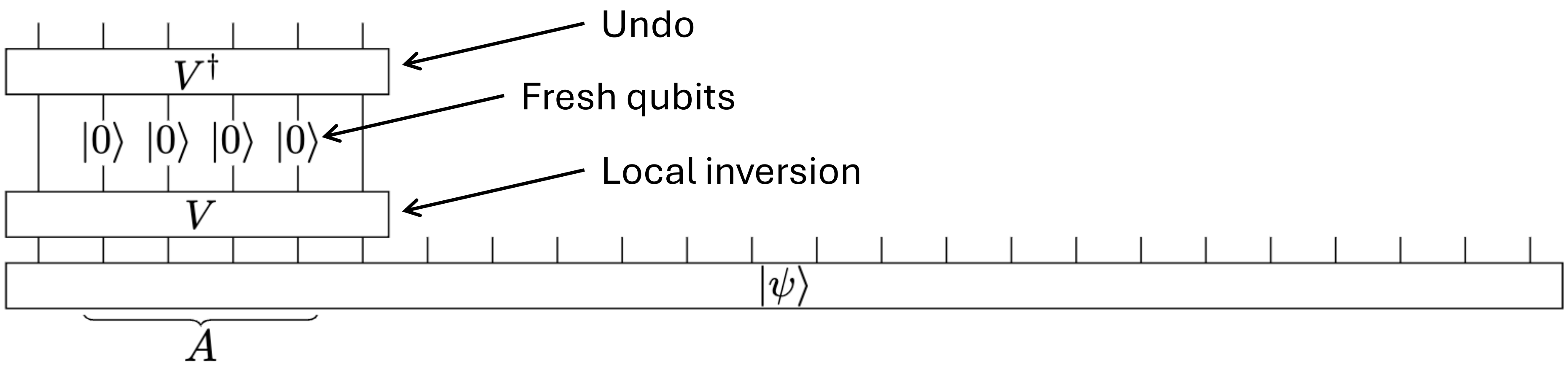
Step 3: Undo the local inversion, by applying $V^\dagger$



Key idea 1: local inversion, *undo*

Observation: the state did not change

$|\psi\rangle =$



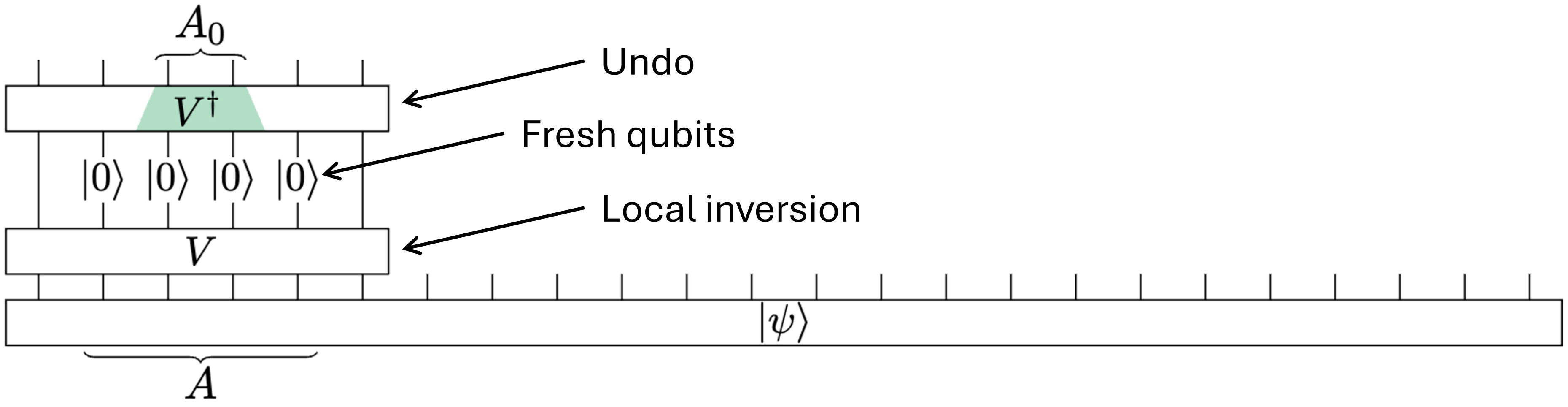
Key idea 1: local inversion, *undo*

Observation: the state did not change

Observation 2: we have learned a small part of the state

- We have learned a circuit for the reduced density matrix on A_0
- Circuit = **backward lightcone**

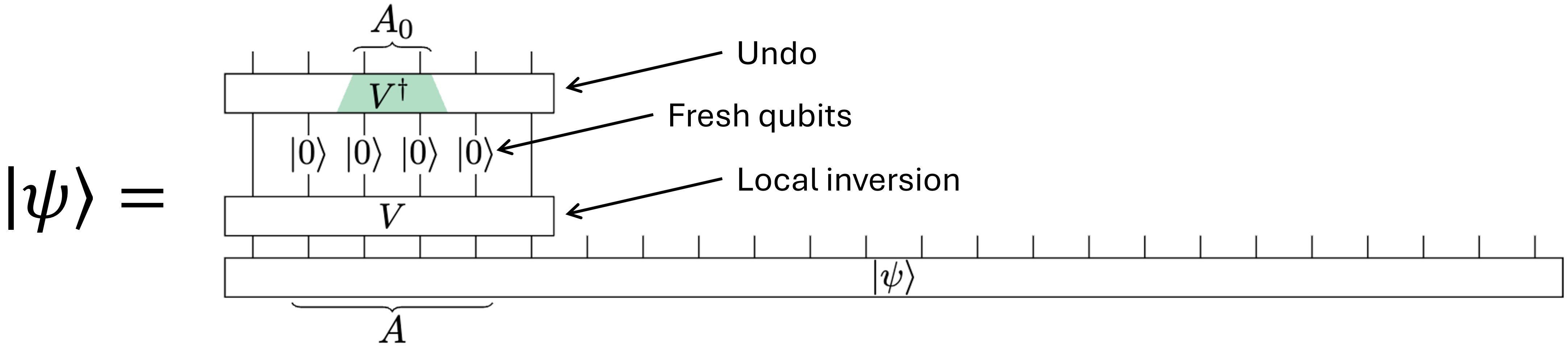
$|\psi\rangle =$



Key idea 1: local inversion, *undo*

Observation: we have reconstructed a small part of the state, without changing the state at all

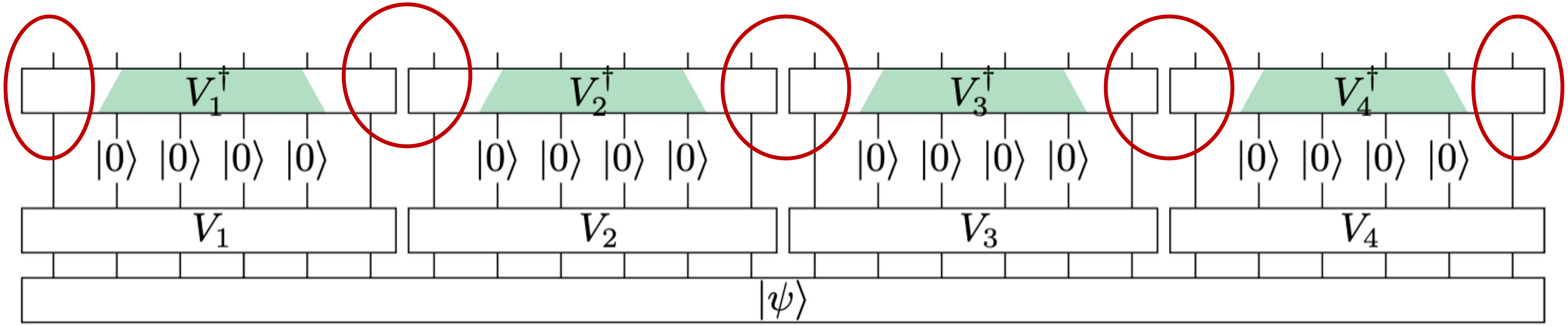
So, why not repeat?



Repeating what we just did

We have reconstructed part of the state
Not enough: what about these regions?

The state did not change! Just repeat by doing another layer

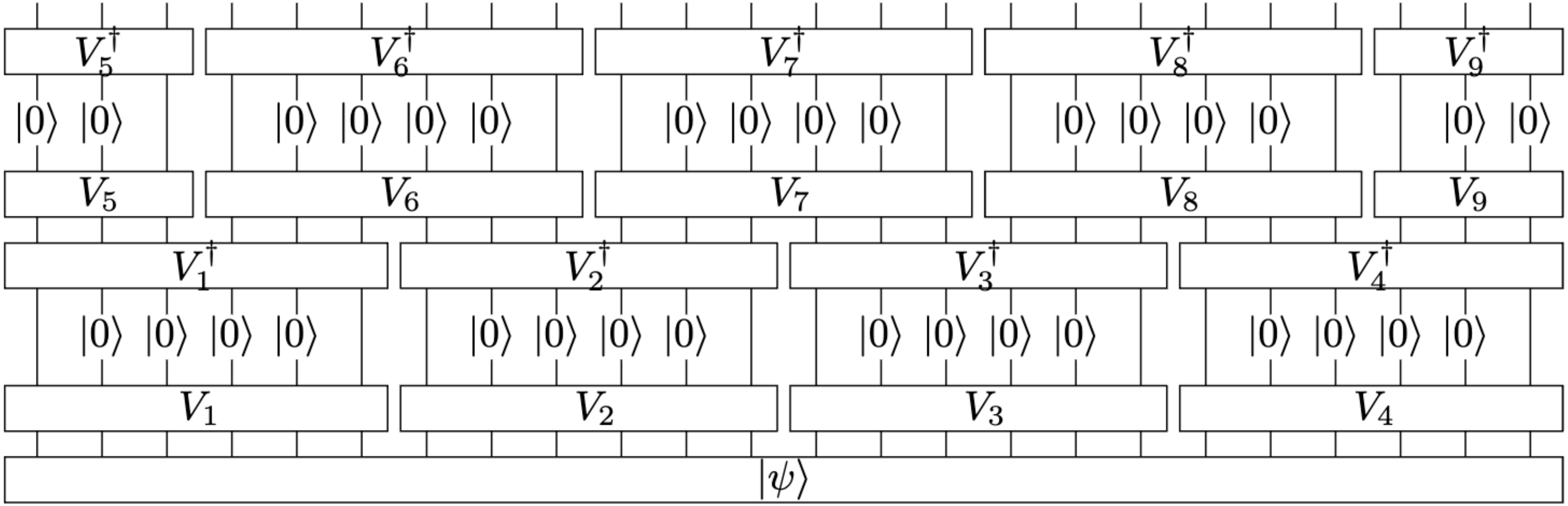
$|\psi\rangle =$ 

Repeating what we just did

Potential issue: the new layer could mess up with what we had earlier

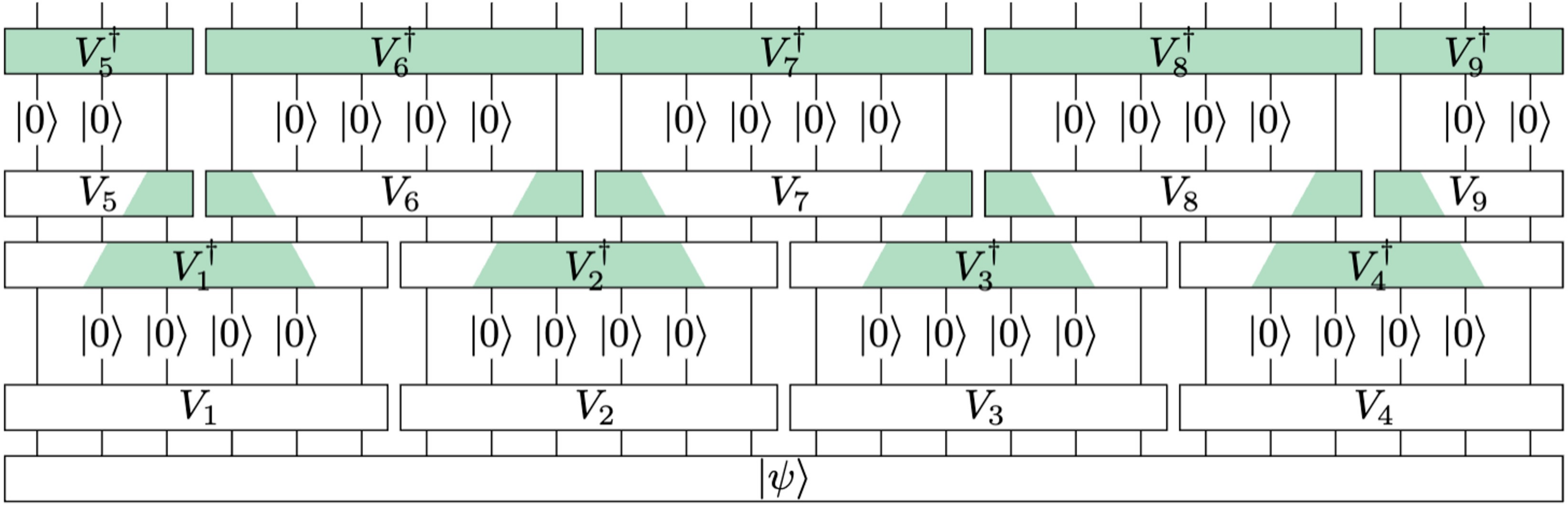
Claim: problem solved; we already reconstructed a circuit for $|\psi\rangle$

$|\psi\rangle =$



Key idea: Reconstruction via backward lightcone

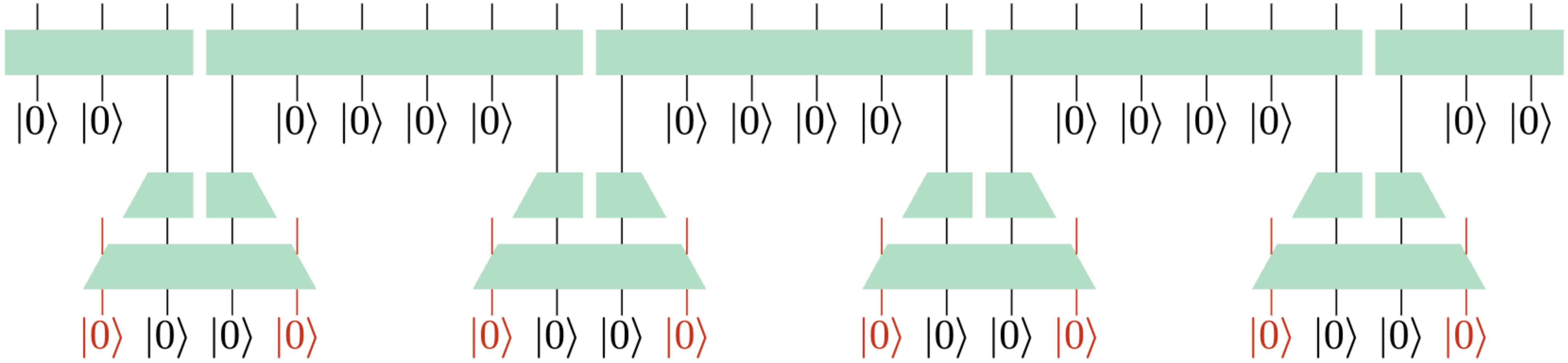
- Reconstruct the output state via backward lightcone of all wires
- This works because backward lightcone stops entirely at fresh qubits

$|\psi\rangle =$ 

Reconstruction via backward lightcone

- Reconstruct the output state via backward lightcone of all wires
- This works because backward lightcone stops entirely at fresh qubits

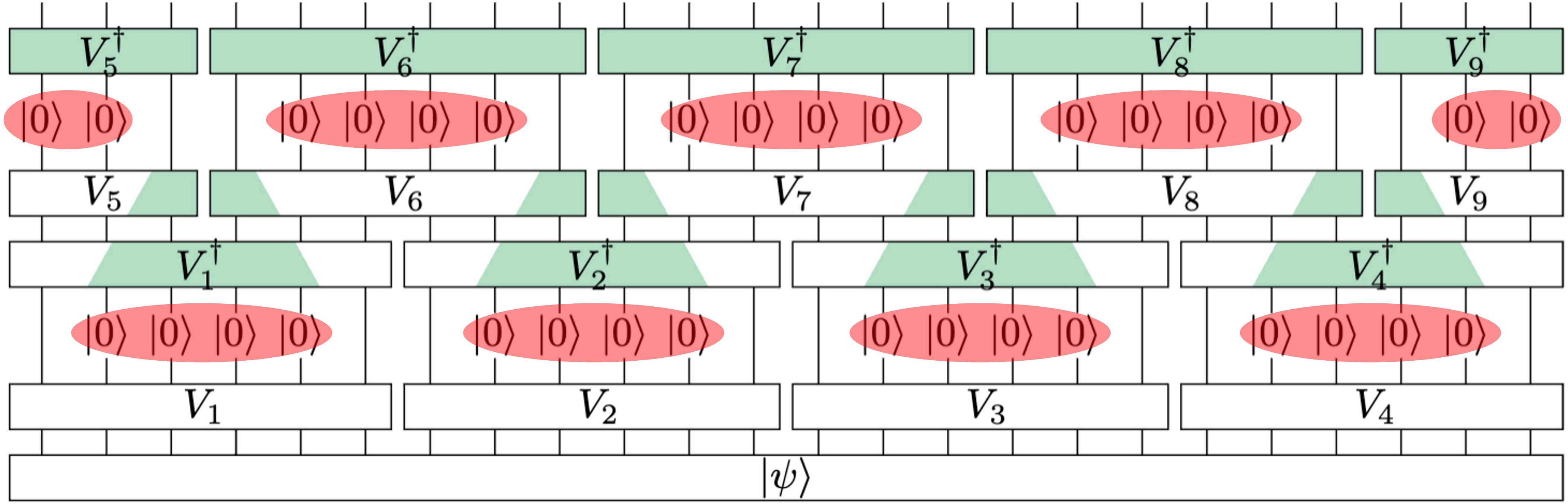
$|\psi\rangle =$



In general, this requires a careful geometric arrangement

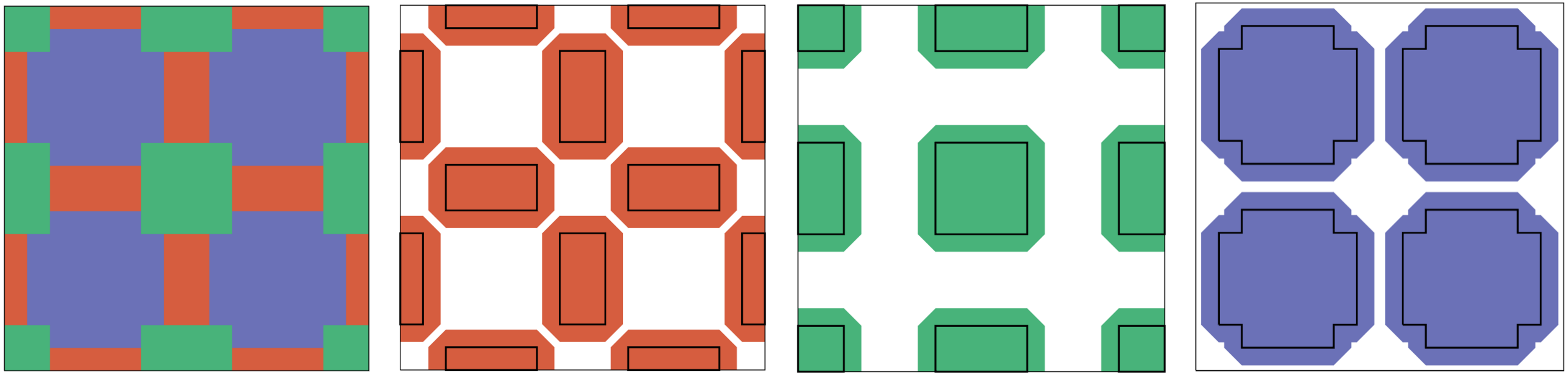
Geometric arrangement: covering scheme

- Divide a lattice into layers of subsets
  - Subsets in the same layer are “sufficiently non-overlapping”
  - Subsets in different layers are “sufficiently overlapping”



Geometric arrangement: covering scheme

- Divide a lattice into layers of subsets
  - Subsets in the same layer are "sufficiently non-overlapping"
  - Subsets in different layers are "sufficiently overlapping"
- Construction: take a lattice coloring, then make every subset larger



Learning phases of matter

- “trivial phase”: quantum states prepared by constant depth circuits on finite dimensional lattice

- Quantum states in the “trivial phase” can be learned efficiently

- Corollary: given an arbitrary state, can efficiently test whether it is in the “trivial phase” (low circuit complexity),

  or “topological phase” (high circuit complexity)

Discussion

More general geometry?

- First algorithm (learning shallow quantum circuits) works for arbitrary or even unknown geometry
  - Recall one-line proof: $U \otimes U^\dagger = (\prod_{i=1}^{n} S_i) \cdot \prod_{i=1}^{n} (U^\dagger S_i U)$
- Second algorithm (learning quantum states prepared shallow circuits) works for any geometry with a good covering scheme
  - We constructed good covering schemes for finite-dimensional lattices

No ancilla qubits?

- Using ancilla qubits is essential in both learning algorithms
- Is it possible to reconstruct the circuit without using any ancilla qubits?

- In [arxiv 2401.10095] we give an algorithm specialized to learning quantum states in 2D, no ancilla qubit assuming finite gate set
  - Key idea: to learn a 2D state it suffices to solve a 1D CSP
  - Probably works for learning quantum circuit in 2D as well?

Outlook

- Toward useful quantum advantage: learning shallow quantum circuits in NISQ algorithms
  - **Two new learning algorithms based on local inversion** that works in simple clean settings; very different from gradient descent
  - Can this be the basis of efficient learning algorithms in more general settings?
    - Need to deal with practical issues such as noise

References

- Learning shallow quantum circuits: arxiv 2401.10095
- Learning quantum states prepared by shallow circuits in polynomial time: