TOWARDS HAMILTONIAN SIMULATION VIA QUBITIZATION ON A TENSOR NETWORK QUANTUM SIMULATOR

PRESENTED BY:
Nathan Fitzpatrick

Senior Research Scientist

Date: 24.10.22

@2022 by Quantinuum. All rights reserved.

CONTRIBUTORS

Iakov Polyak

Software Development

Myself : Quantum Algorithms

NVIDIA cuTensorNet Team

NVIDIA

MOTIVATION

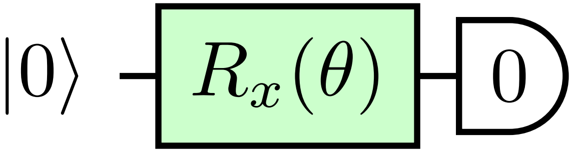
- Hamiltonian Simulation via qubitization has optimal complexity
- Qubitization has opened up a new pathway for algorithms research generally
- The fault tolerant circuit primitives are often given via oracular frameworks
  - Large number of ancillas
  - Complex/ unknown circuits

- Test and build the circuit primitives with pytket for qubitization algorithms such as Hamiltonian simulation

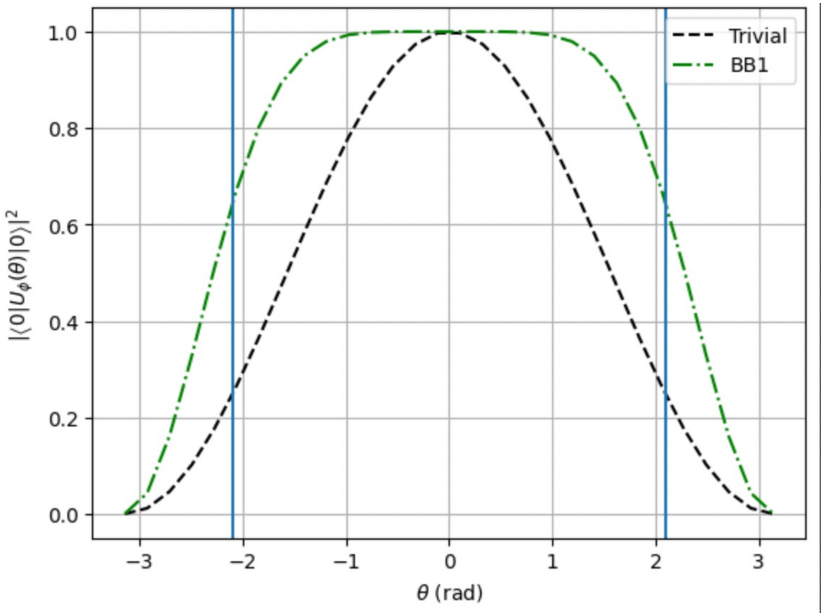
- CuTensorNet simulator + Perlmutter allows us to simulate larger numbers of qubits

QUANTUM SIGNAL PROCESSING

- Single Qubit Example
- Signal rotation



$$\begin{pmatrix} \cos\theta & -i\sin\theta \\ -i\sin\theta & \cos\theta \end{pmatrix}$$



$$|\langle 0|U(\theta)|0\rangle|^2 = \cos^2\theta$$

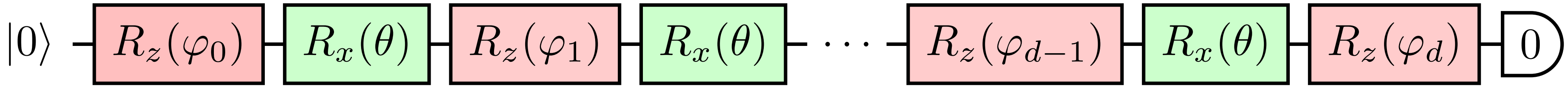
QUANTINUUM

@2022 by Quantinuum. All rights reserved.

Guang Hao Low and Isaac L. Chuang. Quantum 3, 163 (2019

QUANTUM SIGNAL PROCESSING

- Single Qubit Example



$$U_\phi(\theta) = \begin{pmatrix} e^{-i\phi_0} & 0 \\ 0 & e^{i\phi_0} \end{pmatrix} \begin{pmatrix} \cos\theta & -i\sin\theta \\ -i\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} e^{-i\phi_1} & 0 \\ 0 & e^{i\phi_1} \end{pmatrix} \begin{pmatrix} \cos\theta & -i\sin\theta \\ -i\sin\theta & \cos\theta \end{pmatrix} \cdots \begin{pmatrix} e^{-i\phi_{d-1}} & 0 \\ 0 & e^{i\phi_{d-1}} \end{pmatrix} \begin{pmatrix} \cos\theta & -i\sin\theta \\ -i\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} e^{-i\phi_d} & 0 \\ 0 & e^{i\phi_d} \end{pmatrix}$$

- For a given set of $\{\phi_0, ..., \phi_d\}$ we can implement any* Chebyshev polynomial of $a = \cos\theta$

$$U_\phi(\theta) = \begin{pmatrix} P(a) & iQ(a)\sqrt{1-a^2} \\ iQ^*(a)\sqrt{1-a^2} & P^*(a) \end{pmatrix}$$

$$|\langle 0|U_\phi(\theta)|0\rangle|^2 = |P(a)|^2$$

@2022 by Quantinuum. All rights reserved.

Guang Hao Low and Isaac L. Chuang. Quantum 3, 163 (2019)

QUANTUM SIGNAL PROCESSING

Theorem 1 (Quantum Signal Processing)**:** *The QSP sequence $U_{\vec{\phi}}$ produces a matrix that may be expressed as polynomial function of a:*

$$e^{i\phi_0 Z}\prod_{k=1}^{d} W(a)e^{i\phi_k Z} = \begin{bmatrix} P(a) & iQ(a)\sqrt{1-a^2} \\ iQ^*(a)\sqrt{1-a^2} & P^*(a) \end{bmatrix}, \quad (4)$$

for $a \in [-1,1]$, and a $\vec{\phi}$ exists for any polynomials P, Q in a such that:

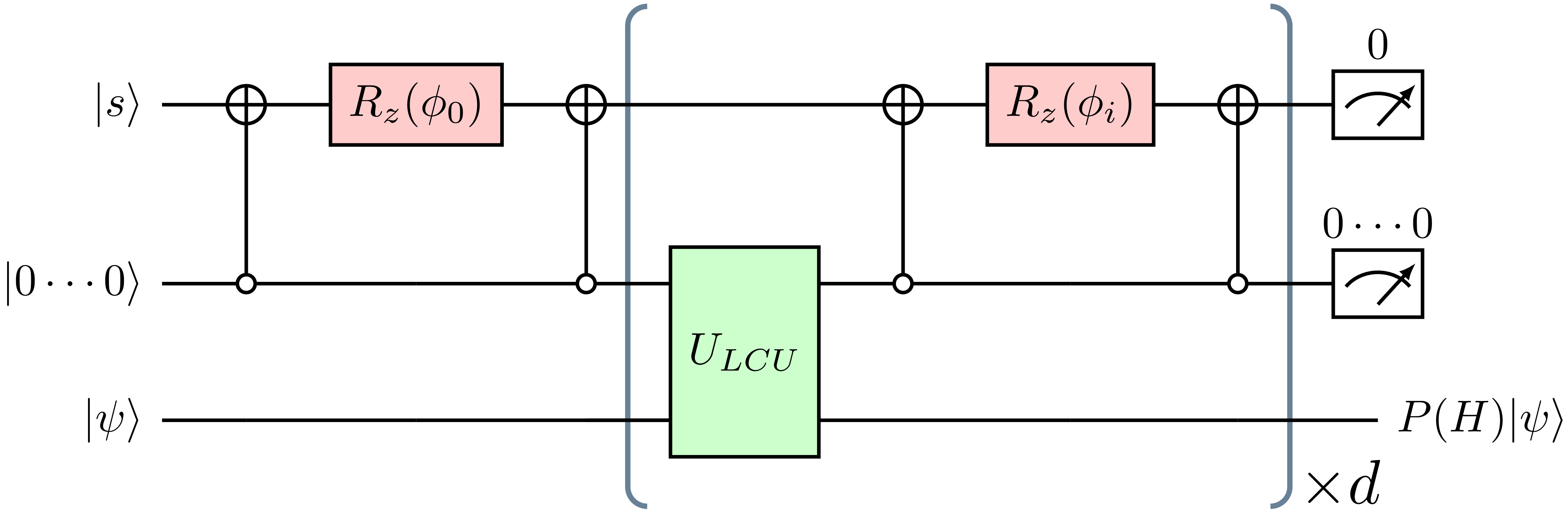
(i) $\deg(P) \leq d$, $\deg(Q) \leq d-1$
(ii) *P has parity d mod 2 and Q has parity $(d-1)$ mod 2*
(iii) $|P|^2 + (1-a^2)|Q|^2 = 1$

- Where. $\{\phi_0, \ldots, \phi_d\}$ can be found efficiently classically

@2022 by Quantinuum. All rights reserved.
J. Martyn PRX QUANTUM 2, 040203 (2021)

QUANTUM SIGNAL PROCESSING

- For a given set of $\{\phi_0, ..., \phi_d\}$ we can implement any* function of H with definite parity

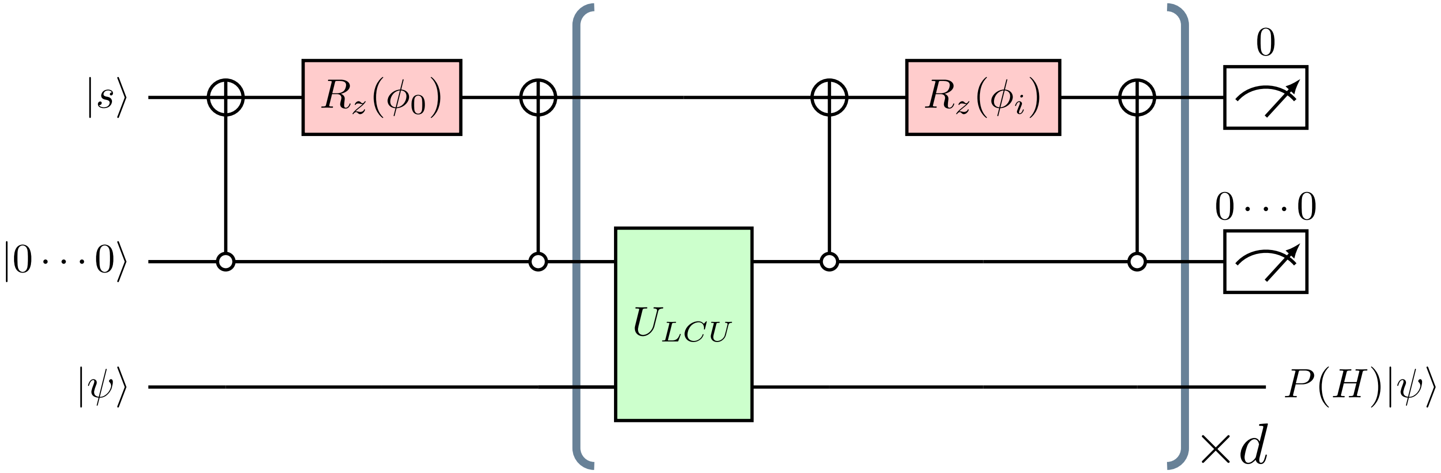


- Larger d gives better accuracy

@2022 by Quantinuum. All rights reserved.

QUANTUM SIGNAL PROCESSING

- For a given set of $\{\phi_0, ..., \phi_d\}$ we can implement any* function of H with definite parity



$$\begin{array}{c} \qquad\quad |0\cdots0\rangle \\ \langle 0\cdots0| \begin{pmatrix} H & * \\ * & * \end{pmatrix} = U_{LCU} \end{array}$$

$$U_\phi(\theta) = \begin{pmatrix} e^{-i\phi_0} & 0 \\ 0 & e^{i\phi_0} \end{pmatrix} \begin{pmatrix} H & \sqrt{1-H^2} \\ \sqrt{1-H^2} & -H \end{pmatrix} \begin{pmatrix} e^{-i\phi_1} & 0 \\ 0 & e^{i\phi_1} \end{pmatrix} \begin{pmatrix} H & \sqrt{1-H^2} \\ \sqrt{1-H^2} & -H \end{pmatrix} \cdots = \begin{pmatrix} P(H) & * \\ * & * \end{pmatrix}$$

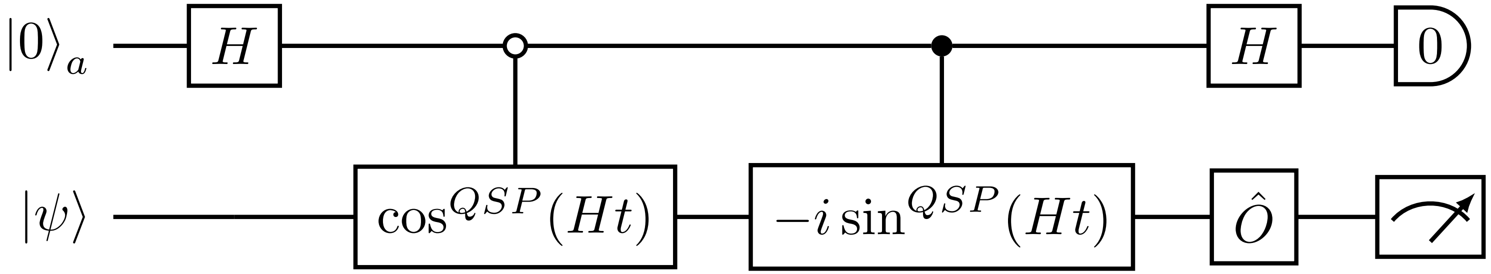
@2022 by Quantinuum. All rights reserved.

LCU = RATE LIMITING STEP

- LCU is the complex circuit primitive
- Aim: Build and compile the circuits for LCU
- Test LCU Contractions on Perlmutter
- Then apply d LCU blocks to implement the full QSP routine

CAVET FOR HAMILTON SIMULATION

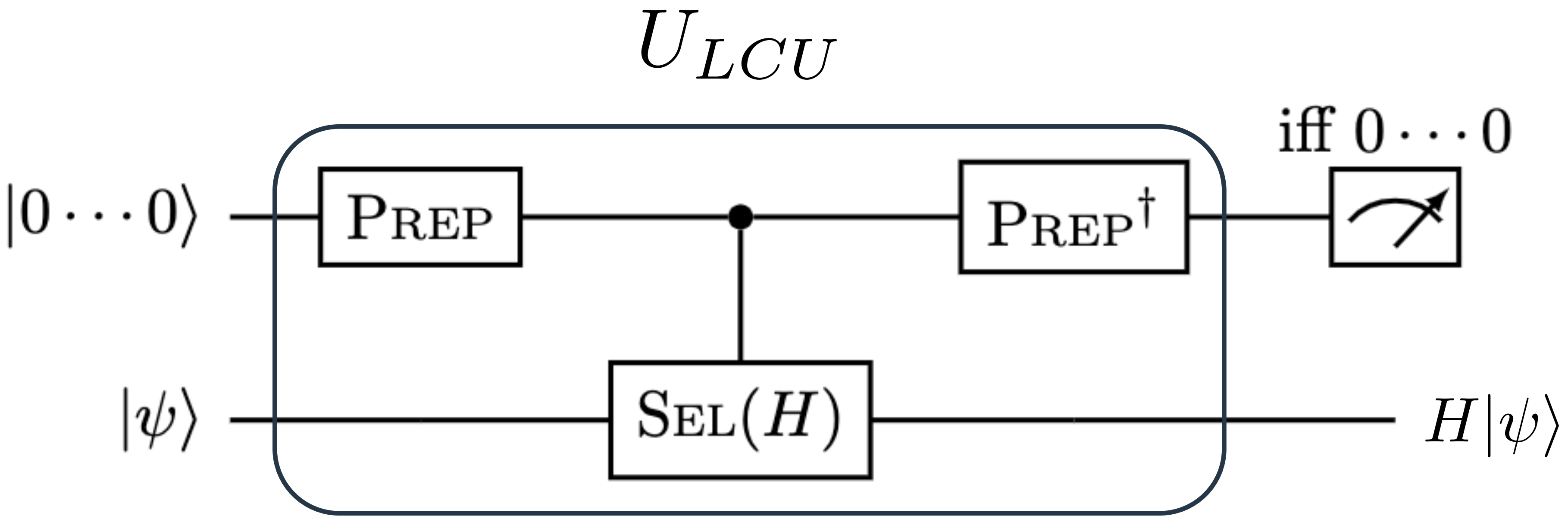
- Due to definite parity e^{-iHt} must be done via $\cos Ht + i \sin Ht$ With LCU.



- For each time we compile a new set of phases $\{\phi_0, ..., \phi_d\}$
- This has been shown to be classically efficient
- $\hat{O}$ is some observable you are simulating over time such as average magnetization etc.

@2022 by Quantinuum. All rights reserved.

Linear Combination of Unitaries



$$\langle 0\cdots 0|U_{LCU}|0\cdots 0\rangle|\psi\rangle$$

$$\langle 0\cdots|\text{PREPARE}\cdot\text{SELECT}(H)\cdot\text{PREPARE}|0\cdots 0\rangle|\psi\rangle \;=\; H|\psi\rangle$$

arXiv:1202.5822

Linear Combination of Unitaries

- *PREPARE*

$$H = \sum_a h_a \hat{P}_a \qquad\qquad ||h|| = \sum_a |h_a|$$

- Encode renormalised Hamiltonian term coefficients onto unique bit strings

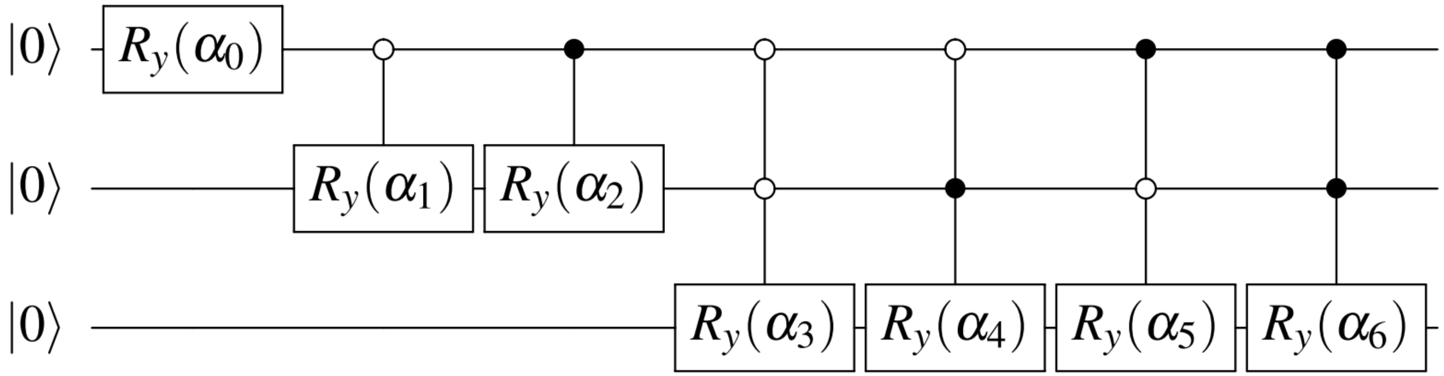
$$|000\rangle \xmapsto{\textsc{Prepare}} \sqrt{\frac{h_0}{||h||}}|000\rangle + \sqrt{\frac{h_1}{||h||}}|001\rangle + \sqrt{\frac{h_2}{||h||}}|010\rangle + \sqrt{\frac{h_3}{||h||}}|011\rangle + \cdots$$

- Arbitrary non fermionic State Preparation
- Motivation to reduce the complexity of the Hamiltonian.
- Negative Hamiltonian coefficients lead to imaginary prepare state coefficients

PREPARE

$$|000\rangle \xmapsto{\textsc{Prepare}} \sqrt{\frac{h_0}{||h||}}|000\rangle + \sqrt{\frac{h_1}{||h||}}|001\rangle + \sqrt{\frac{h_2}{||h||}}|010\rangle + \sqrt{\frac{h_3}{||h||}}|011\rangle + \cdots$$

- Serial with multi controls
- Very deep circuits



- Complex arbitrary state preparation is non trivial.
- Can be absorbed later

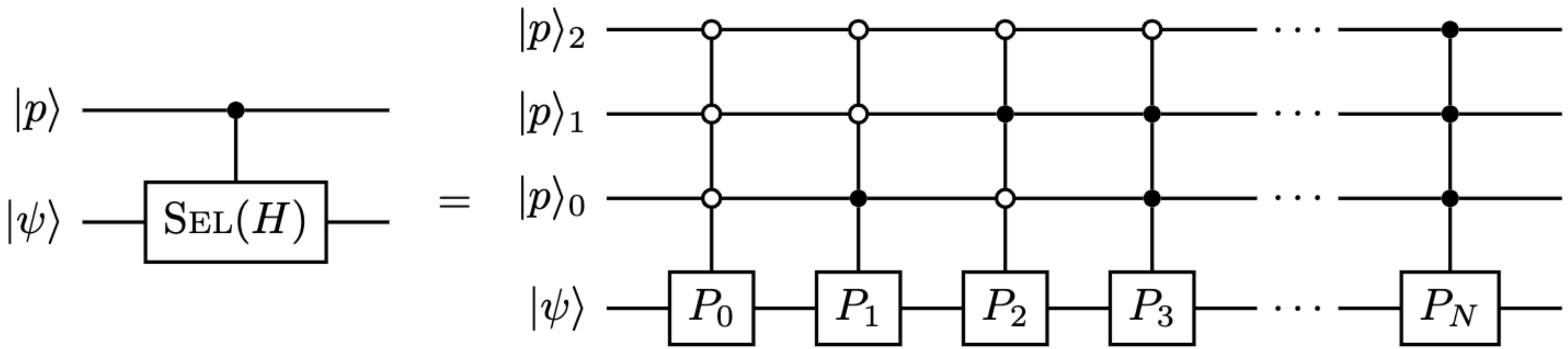
@2022 by Quantinuum. All rights reserved.

Linear Combination of Unitaries

- *SELECT*

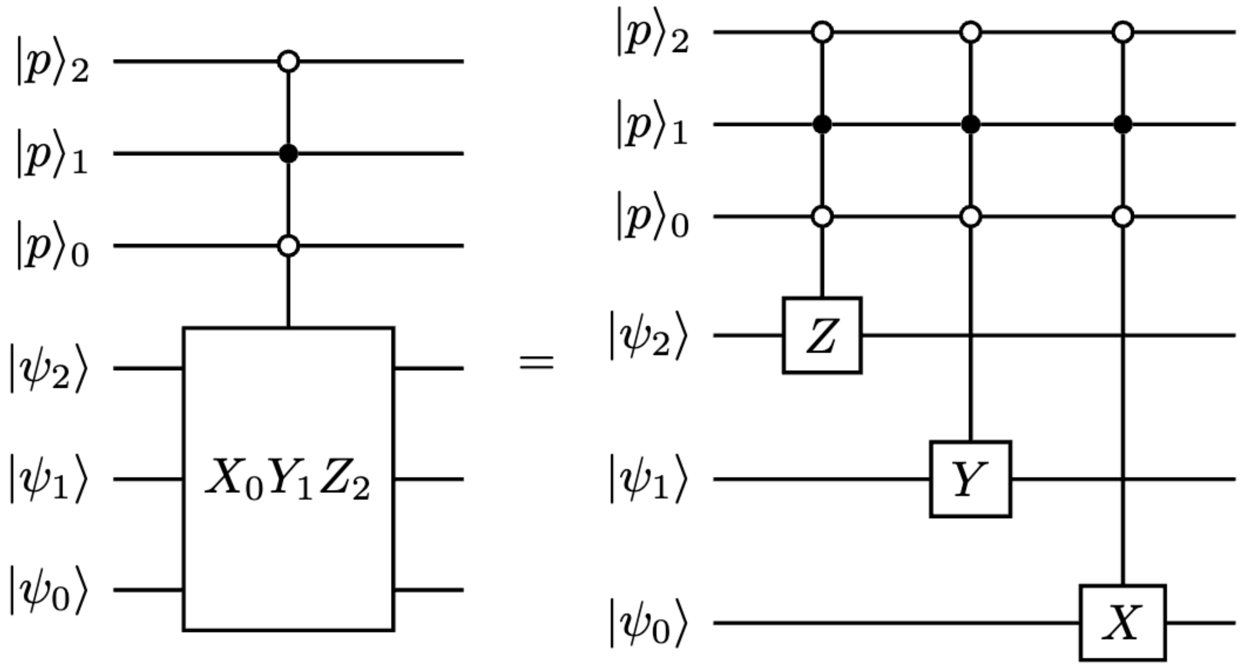
$$H = \sum_a h_a \hat{P}_a$$

- Weighted bit strings applied to Hamiltonian Pauli terms applied $|\psi\rangle$



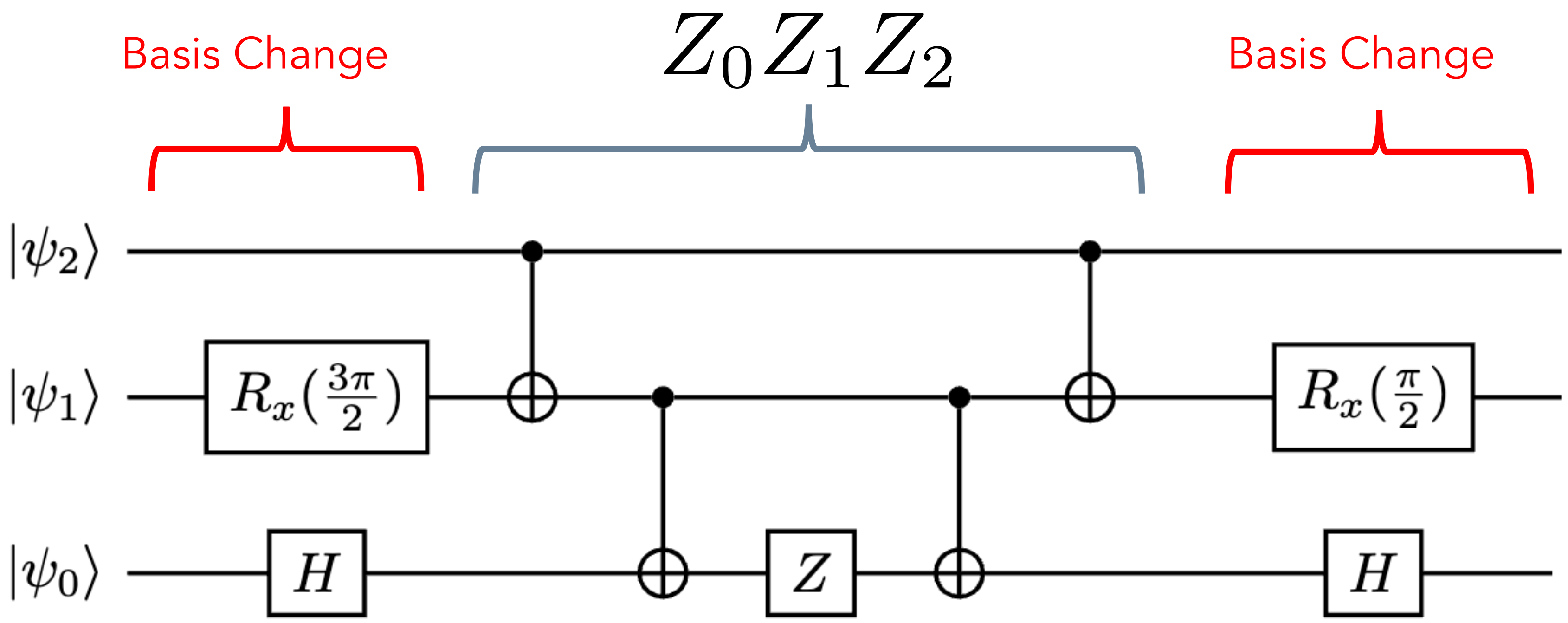
$$|p\rangle|\psi\rangle \xmapsto{\textsc{Select}} \sqrt{\frac{h_0}{||h||}}|000\rangle P_0|\psi\rangle + \sqrt{\frac{h_1}{||h||}}|001\rangle P_1|\psi\rangle + \sqrt{\frac{h_2}{||h||}}|010\rangle P_2|\psi\rangle + \sqrt{\frac{h_3}{||h||}}|011\rangle P_3|\psi\rangle + \cdots$$

Serial Pauli Compilation



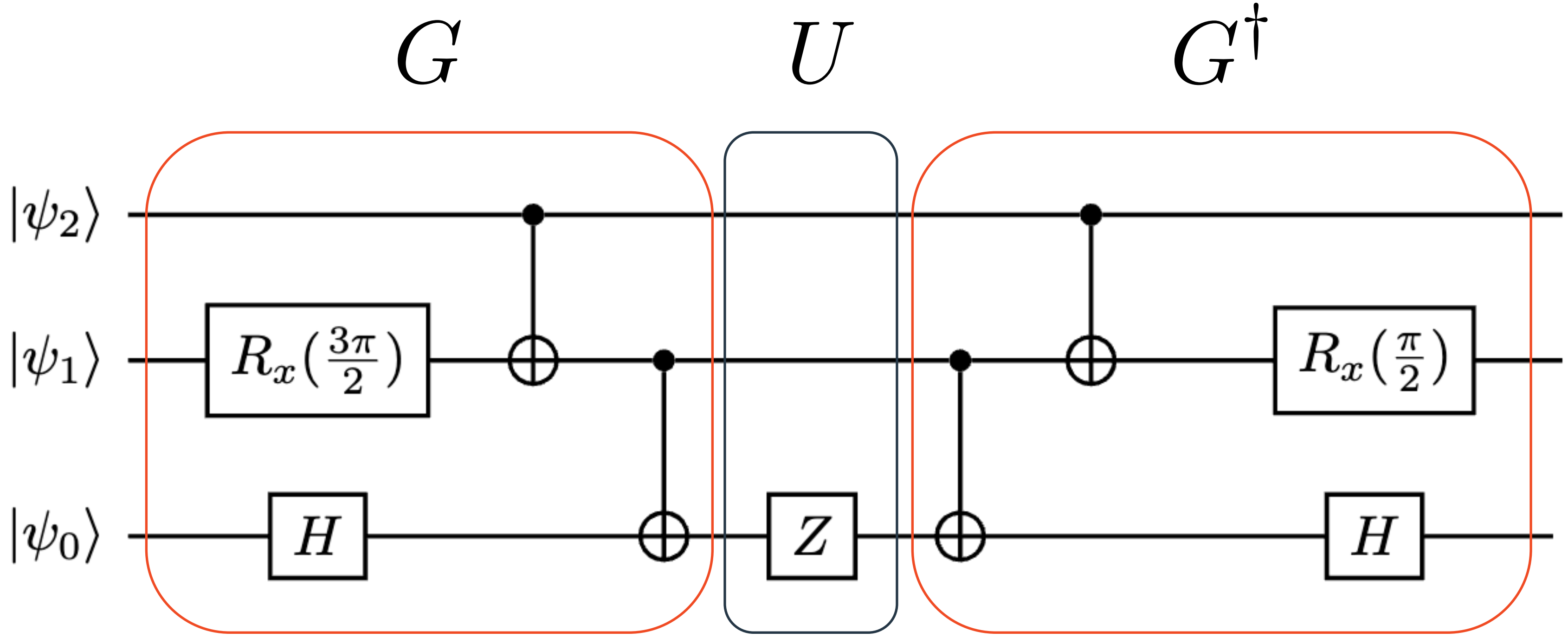
- Most straight forward method
- Can be done with Qcontrol box in pytket
- Leads to many multi controlled gates

Pauli Gadget Select Compilation



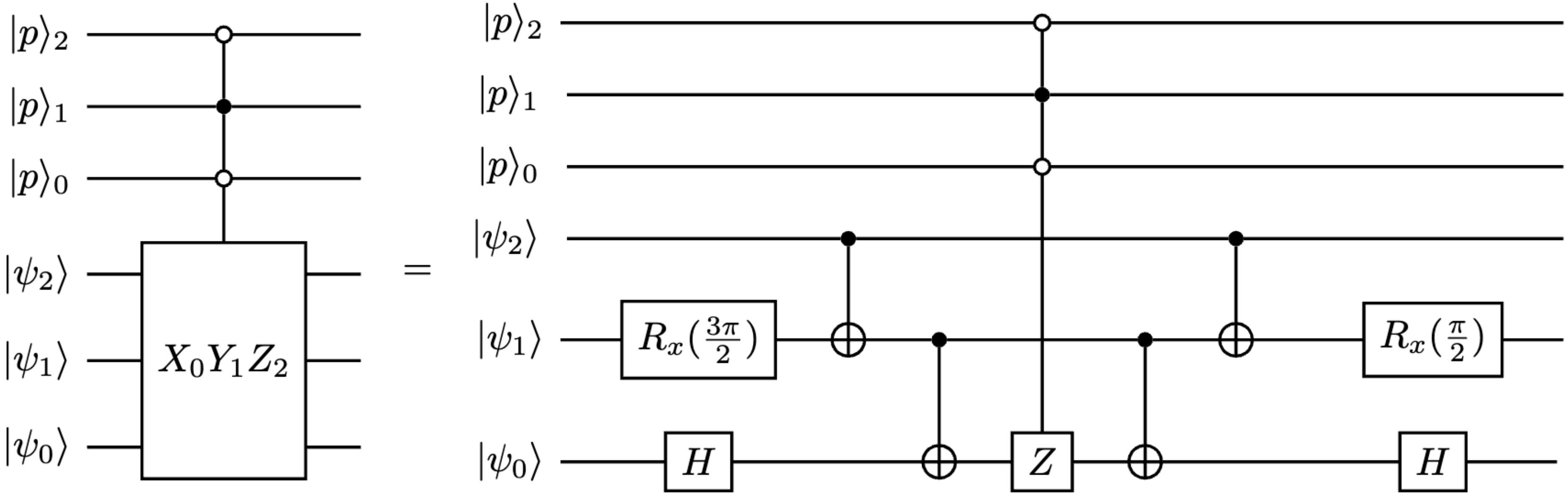
$$Z_2 Y_1 X_0$$

Pauli Gadget Select Compilation



- Similarity transform structure
- $GG^\dagger = 1$ only U needs to be controlled

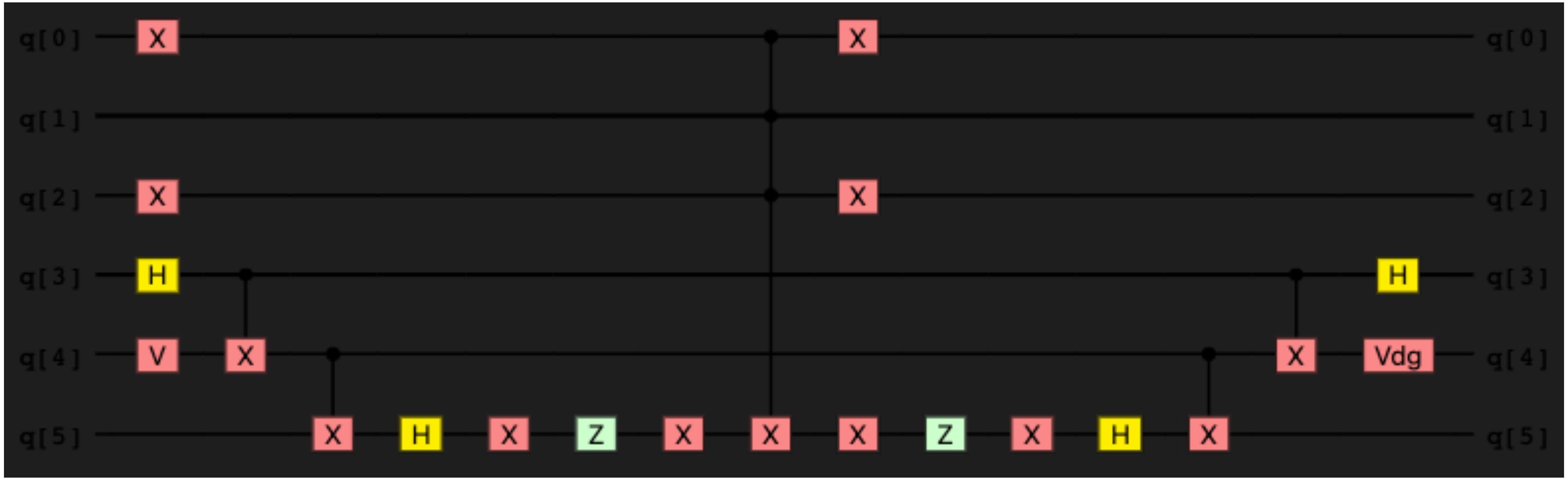
Pauli Gadget Compilation



- Multicontrols decomposed with pytket

Pauli Gadget Select Compilation

- Negative sign absorbed into the pauli gadget using pauli multiplication



Linear Combination of Unitaries

- *PREPARE*†

$$H = \sum_a h_a \hat{P}_a \qquad\qquad ||h|| = \sum_a |h_a|$$

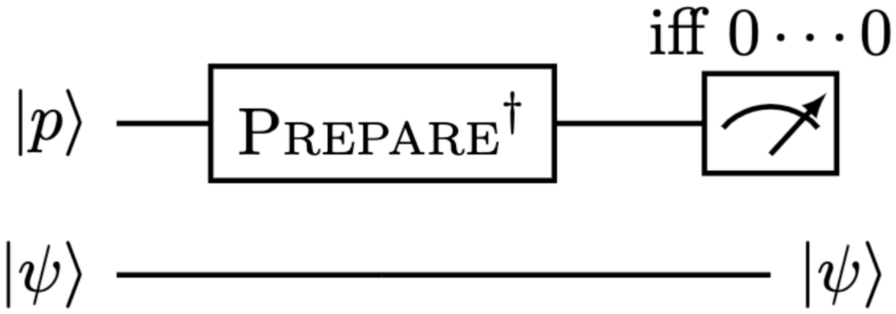
- Rotate weighted Paulis back onto |000>

$$|p\rangle|s\rangle \xmapsto{\text{PREPARE}^{\dagger}} \frac{h_0}{||h||}|000\rangle P_0|\psi\rangle + \frac{h_1}{||h||}|000\rangle P_1|\psi\rangle + \frac{h_2}{||h||}|000\rangle P_2|\psi\rangle \cdots$$

- MEASURING |000> Applies H to $|\psi\rangle$

Linear Combination of Unitaries

- Measure |000>



- Projects onto |0…0> <0…0| block of U_{LCU}

$$\langle 0\cdots 0|\frac{h_0}{||h||}|0\cdots 0\rangle P_0|\psi\rangle + \langle 0\cdots 0|\frac{h_1}{||h||}|0\cdots 0\rangle P_1|\psi\rangle + \langle 0\cdots 0|\frac{h_2}{||h||}|0\cdots 0\rangle P_2|\psi\rangle \cdots = H|\psi\rangle$$

$$\frac{h_0}{||h||}P_0|\psi\rangle + \frac{h_1}{||h||}P_1|\psi\rangle + \frac{h_2}{||h||}P_2|\psi\rangle + \cdots = H|\psi\rangle$$

LCU CIRCUIT BENCHMARKS

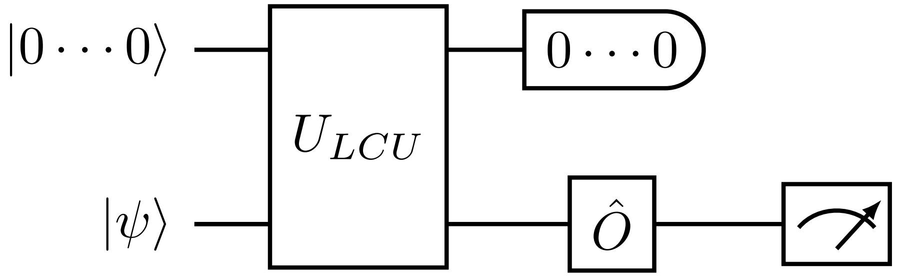
- Hubbard chain and Hydrogen Ring

system	compilation	n_system_qubits	n_ham_terms	n_lcu_qubits	n_lcu_depth	n_gates	l1_norm
2_hubbard_e-2_u2_t-1	SERIAL	4	11	8	947	1320	8
2_hubbard_e-2_u2_t-1	PAULIG	4	11	8	1194	1608	8
3_hubbard_e-2_u2_t-1	SERIAL	6	18	11	4529	6472	13
3_hubbard_e-2_u2_t-1	PAULIG	6	18	11	4754	6966	13
4_hubbard_e-2_u2_t-1	SERIAL	8	25	13	6624	9480	18
4_hubbard_e-2_u2_t-1	PAULIG	8	25	13	6662	9780	18
5_hubbard_e-2_u2_t-1	SERIAL	10	32	15	8830	12638	23
5_hubbard_e-2_u2_t-1	PAULIG	10	32	15	8678	12724	23
6_hubbard_e-2_u2_t-1	SERIAL	12	39	18	17840	25954	28
6_hubbard_e-2_u2_t-1	PAULIG	12	39	18	17338	25548	28
7_hubbard_e-2_u2_t-1	SERIAL	14	46	20	21353	31074	33
7_hubbard_e-2_u2_t-1	PAULIG	14	46	20	20573	30308	33
8_hubbard_e-2_u2_t-1	SERIAL	16	53	22	24723	36010	38
8_hubbard_e-2_u2_t-1	PAULIG	16	53	22	23669	34896	38
9_hubbard_e-2_u2_t-1	SERIAL	18	60	24	28300	41212	43
9_hubbard_e-2_u2_t-1	PAULIG	18	60	24	26968	39734	43
10_hubbard_e-2_u2_t-1	SERIAL	20	67	27	50696	73774	48
10_hubbard_e-2_u2_t-1	PAULIG	20	67	27	48478	70852	48
2_h_ring_3.0A_sto3g	SERIAL	4	15	8	1676	2352	1.56661938
2_h_ring_3.0A_sto3g	PAULIG	4	15	8	1663	2246	1.56661938
4_h_ring_3.0A_sto3g	SERIAL	8	105	15	120550	181612	4.75500962
4_h_ring_3.0A_sto3g	PAULIG	8	105	15	76291	111982	4.75500962
6_h_ring_3.0A_sto3g	SERIAL	12	703	22	2411680	3845726	12.3424871
6_h_ring_3.0A_sto3g	PAULIG	12	703	22	1159480	1720742	12.3424871

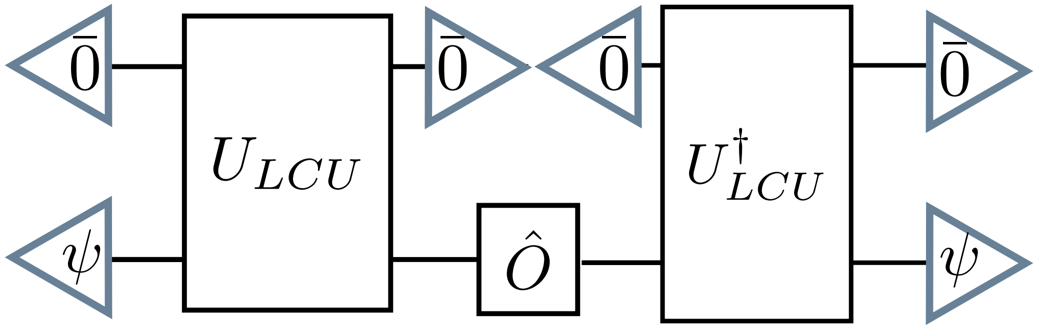
@2022 by Quantinuum. All rights reserved.

LCU TENSOR NETWORK MEASUREMENT

- Circuit requires post selection



- Need to therefore cap the tensor network for the given qubit



- Unphysical – purely for benchmarking LCU contractions for when they will eventually be used in QSP

@2022 by Quantinuum. All rights reserved.

LCU CIRCUIT BENCHMARKS

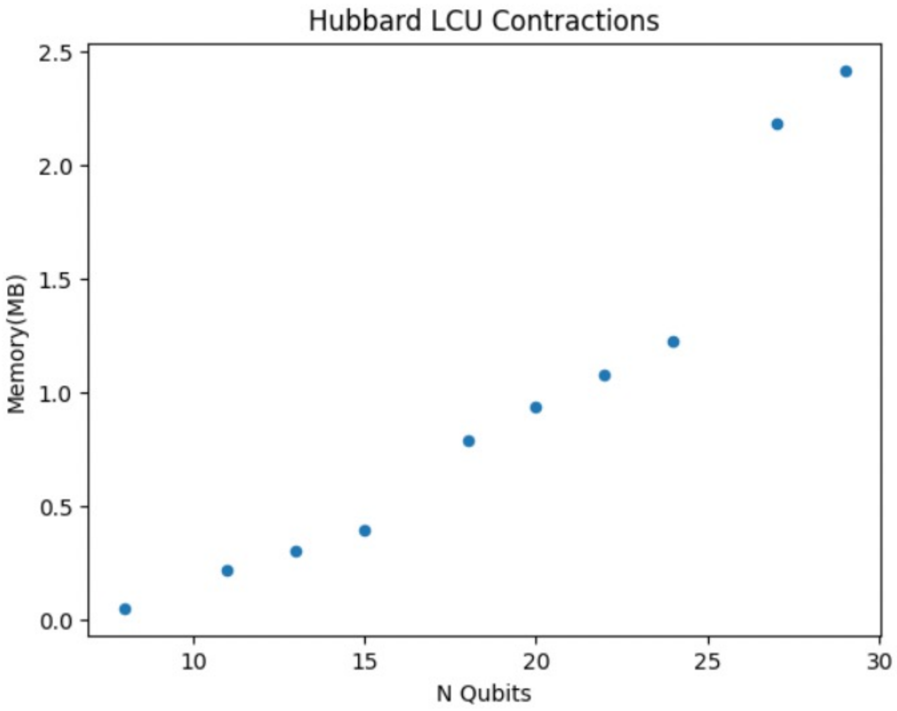
- Hubbard chain LCU Contractions

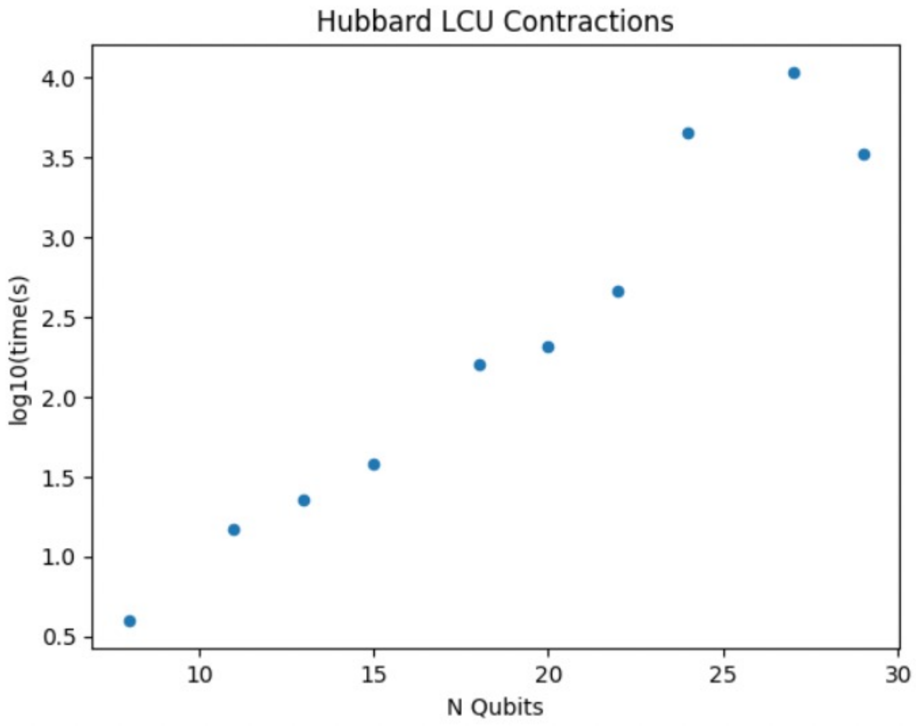
system	n_gates	depth	tensors	tot_n_qubits	system_qubits	memoryGB	memory(MB)	calc_time(s)
2_hubbard_gadget	1666	1214	3332	8	4	5.01E-05	0.051322937	3.97959185
3_hubbard_gadget	7084	4770	14168	11	6	0.000211738	0.216819763	14.8165562
4_hubbard_gadget	9922	6690	19844	13	8	0.000296406	0.303520203	22.9060276
5_hubbard_gadget	12890	8704	25780	15	10	0.000384949	0.394187927	38.3699226
6_hubbard_gadget	25834	17380	51668	18	12	0.000770859	0.789360046	158.725572
7_hubbard_gadget	30634	20629	61268	20	14	0.000914	0.935935974	209.390789
8_hubbard_gadget	35256	23727	70512	22	16	0.001051836	1.077079773	462.465788
9_hubbard_gadget	40124	27026	80248	24	18	0.001197003	1.225730896	4532.92379
10_hubbard_gadget	71416	48544	142832	27	20	0.002129726	2.180839539	10837.8559
11_hubbard_gadget	79112	53799	158224	29	22	0.002359174	2.415794373	3354.59784

@2022 by Quantinuum. All rights reserved.

LCU CIRCUIT BENCHMARKS

- Hubbard chain LCU Contractions





- Exponential in Contraction time with system size
- Linear in memory (for each qubit regime in the prepare register) with system size

@2022 by Quantinuum. All rights reserved.

CONCLUSIONS AND NEXT STEPS

Done

- Built pytket -> cuTensorNet converter with post selection
- Implemented LCU contractions on Perlmutter

To Do

- Apply LCU d times to implement quantum signal processing for Hamiltonian simulation
- Look into further compilation strategies that leverage more ancillas which may be better suited for tensor network contraction
- Does repeating circuit structure nature allow for simpler contraction path

Linear Combination of Unitaries

- Projects onto |0…0> <0…0| block of U_{LCU}

$$\langle 0 \cdots 0| \frac{h_0}{||h||} P_0 + \frac{h_1}{||h||} P_1 + \frac{h_2}{||h||} P_2 + \frac{h_3}{||h||} P_3 + \cdots |000\rangle |\psi\rangle = H|\psi\rangle$$

- Post selection success probability

$$P_{success} = \langle \psi | \langle 0 \cdots 0| \frac{H^\dagger}{||h||} \frac{H}{||h||} |0 \cdots 0\rangle |\psi\rangle$$

$$= \frac{1}{||h||^2} \langle \psi | H^\dagger H || \psi \rangle$$

Ralli, Love, Tranter, Coveney - PHYSICAL REVIEW RESEARCH 3, 033195 (2021)