IEEE Transactions on Quantum Engineering

Received 5 December 2023; revised 26 April 2024; accepted 11 May 2024; date of publication 16 May 2024; date of current version 25 June 2024.

Digital Object Identifier 10.1109/TQE.2024.3402085

Harnessing the Power of Long-Range Entanglement for Clifford Circuit Synthesis

WILLERS YANG AND PATRICK RALL
IBM Quantum, MIT-IBM Watson AI Lab, Cambridge, MA 02142 USA

Corresponding author: Willers Yang (e-mail: willers@uchicago.edu).

This work was supported by IBM Quantum.

ABSTRACT In superconducting architectures, limited connectivity remains a significant challenge for the synthesis and compilation of quantum circuits. We consider models of entanglement-assisted computation where long-range operations are achieved through injections of large Greenberger–Horne–Zeilinger (GHZ) states. These are prepared using ancillary qubits acting as an "entanglement bus," unlocking global operation primitives such as multiqubit Pauli rotations and fan-out gates. We derive bounds on the circuit size for several well-studied problems, such as CZ circuit, CX circuit, and Clifford circuit synthesis. In particular, in an architecture using one such entanglement bus, we give a synthesis scheme for arbitrary Clifford operations requiring at most $2n + 1$ layers of entangled state injections, which can be computed classically in $O(n^3)$ time. In a square-lattice architecture with two entanglement buses, we show that a graph state can be synthesized using at most $\lceil \frac{1}{2}n \rceil + 1$ layers of GHZ state injections, and Clifford operations require only $\lceil \frac{3}{2}n \rceil + O(\sqrt{n})$ layers of GHZ state injections.

INDEX TERMS Clifford circuits, Greenberger–Horne–Zeilinger (GHZ) states, long-range entanglement, quantum circuit synthesis.

I. INTRODUCTION

Unlike classical random access memories where direct access to arbitrary bits comes at a low cost, quantum operations across nonadjacent qubits often incur significant additional overhead. One common resolution is to use SWAP gates to bring qubits to adjacent positions. However, when we use SWAP gates for qubit routing, we may suffer an overhead in depth that is linear in the number of qubits. While this is not concerning for exponential speedups in theory, in practice, this overhead could render quantum algorithms with only mild polynomial speedups useless and otherwise dull the quantum competitive edge.

Another solution is to use long-range entanglement to implement nonlocal operations. Local measurements with feedforward corrections allow us to prepare quantum states with long-range entanglement in constant depth [6], [7]. Using gate teleportation and similar techniques, these states can be used as a resource to implement long-range two-qubit gates and even global n-qubit gates [2]. This observation suggests the following trade: sacrifice a constant fraction of the qubits to act as an "entanglement bus" and obtain a certain flavor of all-to-all connectivity in exchange.

These techniques are widely considered in surface code architectures, especially lattice surgery [2], which reformulates Clifford + T circuits in terms of ancilla-assisted multiqubit Pauli rotations. Works on surface code routing leverage constant-depth preparation of Bell states to facilitate long-range CNOT gates [7], [8]. Other models leverage Hamiltonian time evolution to implement certain n-qubit gates and discuss their utility toward implementing permutations [9] and Clifford operations [10], [11], [12]. It is also well known that certain families of interesting quantum states in physics are easy to prepare using measurement and feedback [13], [14]. Using entangled states as a resource for computation is a central idea in the field of measurement-based quantum computation [15], whose techniques enable us to trade circuit depth for circuit width.

Previous works on surface code compilation have performed numerical studies either on the speed of implementing fixed sequences of CNOT gates [7] or on asymptotic bounds on the implementation of permutations using entanglement routing [8]. Our work takes inspiration from these proposals while also exploiting the particular structure of the Clifford group and studying the leading coefficient in the

TABLE 1 Comparison of Best Known Upper and Lower Bounds on Circuit Depth for Graph States and Clifford Operations

Model	Depth Metric	State Synthesis	Clifford Synthesis
LNN	CNOT	– $\leq 2n+2$ [2][1]	$\geq 2n+1$ [1] $\leq 7n-4$ [3]
All-to-all	CNOT	$\leq \frac{n}{2}+O(\log^2(n))$ [4]	$\leq 2n+O(\log^2(n))$ [4]
Linear GHZ Bus	GHZ injection	$\geq$minrank(G) (clique flips) $\leq$minrank$(G)+1$ (Proposition1)	$\geq 0.648n-2$ (Proposition 9) $\leq 2n+1$ (Corollary1, [5])
Dual Snake	GHZ injection	$\leq \lceil \frac{n}{2} \rceil + 1$ (Proposition4)	$\leq \lceil \frac{3}{2} \rceil n + O(\sqrt{n})$ (Corollary2)

The linear GHZ bus and dual snake models are proposed in this work and explained in Section II.

synthesis performance. This approach allows us to incorporate more sophisticated optimizations and achieve highly parallelized circuits with competitive upper bound guarantees using a smaller fraction of ancillary qubits.

Efficient circuit synthesis of Clifford operations is not just central to the implementation of fault-tolerant quantum algorithms, but it is also extensively studied in noisy intermediate-scale quantum (NISQ) models. In particular, using single-qubit gates and two-qubit entangling gates, such as the CNOT and CZ gates, we can show that arbitrary Clifford circuits can be synthesized using at most $7n-4$ layers of two-qubit gates with linear nearest neighbor (LNN) architecture [3] and at most $2n+O(\log^2(n))$ layers of two-qubit gates with all-to-all connectivity [4]. When we allow global operations, an algorithm exists that computes the optimal decomposition of a Clifford operation into Pauli rotations, achieving a worst case gate count of $2n+1$ [5]. Finally, in architectures that allow a more powerful global tunable gate, Clifford operations may require only constant depth [12].

The rest of this article is organized as follows. First, we will describe the Greenberger–Horne–Zeilinger (GHZ) bus models in detail in Section II and relate them to other proposals in prior work. Then, in Section III, we will present several optimization techniques to efficiently synthesize various classes of Clifford circuits, starting with CZ circuits, CX circuits, and Hadamard-free circuits in Sections III-A–III-C, respectively. In particular, our construction for CZ circuits achieving depth $\lceil \frac{1}{2}n \rceil + 1$ in a model with two GHZ buses can also be applied to the synthesis of graph states. Then, combining these optimization techniques, we arrive at our main results on Clifford synthesis in Section III-D: first, in a model with one GHZ bus, we present a simplified construction achieving the optimal GHZ injection depth guarantee of $2n+1$ in [5]; second, in a more powerful model with two GHZ buses with square lattice connectivity, we present a highly parallelized construction achieving GHZ injection depth $\lceil \frac{3}{2} \rceil n + O(\sqrt{n})$ for Clifford synthesis. Our results are summarized in Table 1, with some lower bound derivations deferred to the Appendix.

II. MODELS

Our results are chiefly inspired by surface code architectures, in which the allocation of "entanglement bus" qubits to facilitate long-range interactions is common in several works [2], [7]. Rather than studying the capabilities of a complex architecture for large quantum circuits in practice, we design simplified models to capture *only* the impact of GHZ state injection on an architecture with otherwise poor connectivity. We expect improved connectivity to be the primary benefit of GHZ state injection, and this architecture lets us quantify the improvement.

In the rest of this section, we define the models we consider in Section II-A and describe in detail the operation primitives enabled, as well as the cost metric we consider. We also include a comparison of our model to prior work in Section II-B.

A. ARCHITECTURE DETAILS

First, let us define the GHZ bus architecture enabling a set of k-qubit Clifford operations as primitive logical operations.

Definition 1: GHZ bus architectures enable a set of gates acting on n qubits. With the qubits enumerated $1, \ldots, n$, the operations are as follows:

1) all single-qubit gates;
2) k-qubit gates acting on k adjacent qubits $i, \ldots, i+(k-1)$ from the following families:
   a) *CNOT fan-out:* if a control qubit is $|1\rangle$, apply X to any subset of the other k qubits;
   b) *Pauli rotation:* for any phase angle ϕ and any multiqubit Pauli operator P supported on k qubits, apply the unitary $\exp(i\phi P)$.

Each of the k-qubit logical operations can be implemented with constant overhead given a GHZ resource state, as presented in Fig. 1(c) and (d). Another primitive that can be implemented is the k-qubit Pauli measurement; however, since our synthesis schemes rely only on Pauli rotations and CNOT fan-outs, we do not include Pauli measurements in our definition. A *GHZ state injection* refers to the consumption of a GHZ state to implement a k-qubit logical operation. The correctness proof is given using ZX calculus in Fig. 2. Since each operation consumes on the GHZ state, we believe that these families of operations equivalently capture the power of GHZ state injection. Certainly, it is easy to see how to prepare a single GHZ state using CNOT fan-out. We can also prepare a GHZ state by applying $\exp(i\frac{\pi}{4}Y^{\otimes n})$ to $|0^n\rangle$ or by measuring the $X^{\otimes n}$ observable on $|0^n\rangle$ and applying a Pauli correction. Interconversions between the operations are less simple, and circuits achieving these are given in Fig. 3. We find that to transform one of these operations into any of the other two, an additional ancilla qubit is required. This makes sense for Pauli measurements since they require an additional degree of freedom to be measured in order to avoid damaging the coherence of the input state. However, the smallest circuit

IEEE Transactions on Quantum Engineering

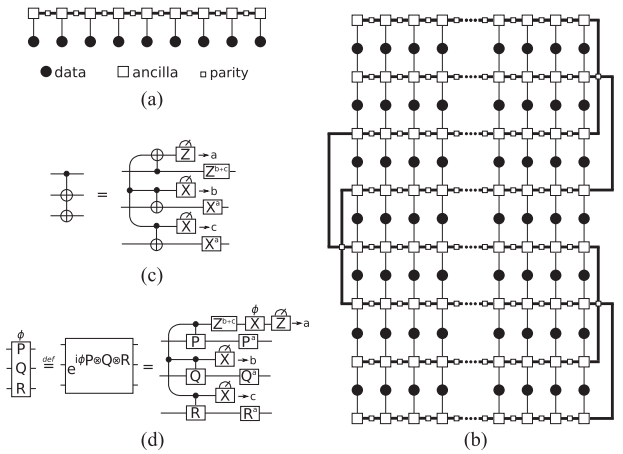

FIGURE 1. Architectures and primitive operations considered in this article. (a) LNN architecture with a "GHZ bus": a "rail" of ancilla qubits reserved for the preparation of GHZ states (see Fig. 4). (b) "Dual snake" architecture compatible with a square lattice of qubits featuring two intertwining GHZ buses. In the limit of many qubits, only about half of the chip area is dedicated to ancillae. This architecture permits the parallelization of two layers of primitive operations, provided that they act on disjoint sets of qubits. (c) Implementation of a fan-out gate using a GHZ state prepared on the GHZ bus. (d) Implementation of a Pauli rotation gate via injection of a GHZ state.

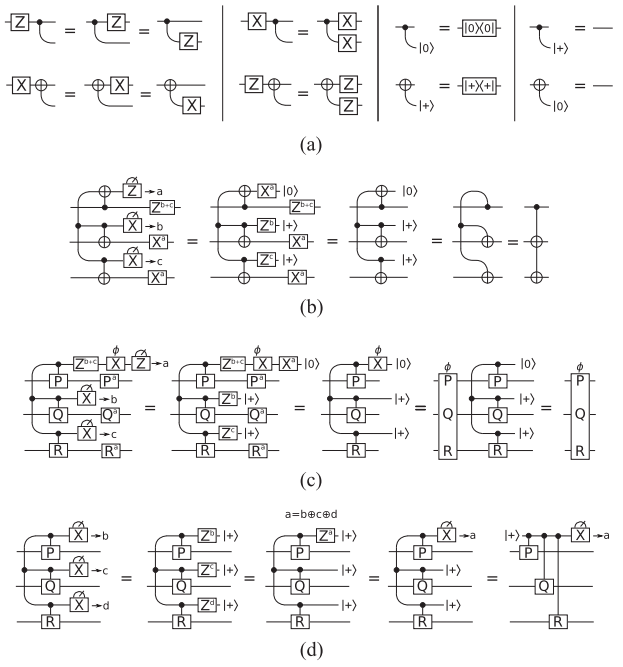

FIGURE 2. Derivation of GHZ state injection circuits using ZX calculus [16]. (a) ZX calculus identities. (b) Derivation: fan-out. (c) Derivation: Pauli rotation. (d) Derivation: Pauli measurement.

without an extra ancilla for implementing fan-out requires two Pauli rotations: one on all the qubits, and on all but the target. Even with the additional ancilla, the synthesis of fan-out gates demands an additional CNOT gate. Despite these limitations, there is plenty of evidence that these three circuit primitives have roughly the same capabilities up to constant factors.

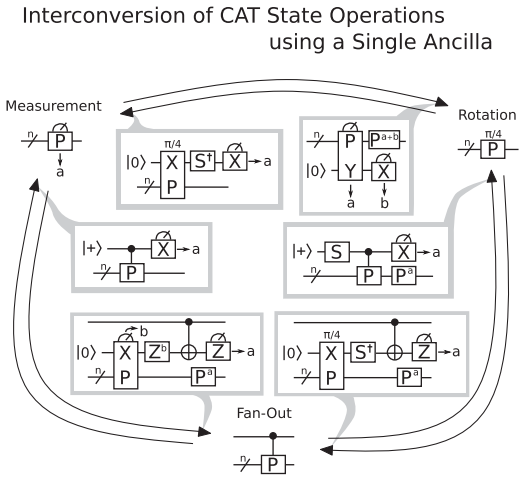

FIGURE 3. Circuits for interconversion of the three GHZ-state-enabled *n*-qubit gates considered in this article: Pauli measurement, Pauli rotation, and fan-out. All of these conversions require an ancilla qubit, and the synthesis of fan-out requires an additional CNOT gate. Otherwise, this is evidence that these three operations have roughly the same power.

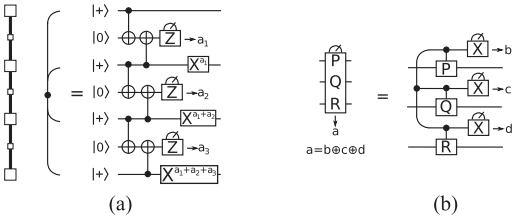

FIGURE 4. (a) Preparation of a GHZ state on the ancillae of the GHZ bus using parity check qubits, as well as two CNOT layers and a mid-circuit measurement. (b) GHZ state injection implementation of another primitive not leveraged in our work: measurement of an *n*-qubit Pauli observable.

The GHZ bus can be implemented efficiently with the instruction set of both fault-tolerant architectures on surface code logical qubits and NISQ architectures on physical qubits. In a surface code architecture, a large rectangular ancilla patch can be prepared to implement multiqubit Pauli measurements in a single code cycle [2]. On physical qubits where two-qubit CNOTs are the native, a GHZ state can be synthesized using a constant depth quantum circuit with two layers of nearest neighbor CNOT gates and one measurement, as presented in Fig. 4(a).

We use circuit depth in terms of *GHZ state injection layers* to quantify the cost of implementing a set of operations on the GHZ bus architecture. First, we note that whenever operations act on nonoverlapping ranges of qubits $l_1 \ldots r_1$ and $l_2 \ldots r_2$ such that $r_1 < l_2$, they can be performed simultaneously. A layer of parallel gates consisting entirely of k-qubit gates is called a *GHZ state injection layer*. We consider single-qubit gates to be free: such approximations are often made for synthesis tasks both in the NISQ setting where the dominant cost comes from implementing entangling gates [3], [4], [17] and in the fault-tolerant setting

where single-qubit Clifford gates can either be implemented transversally or be tracked in software [2], [18].

The relative cost of the nearest neighbor two-qubit gates and the GHZ state injections depends on the implementation of the model. Recall that in a surface code architecture based on lattice surgery, CNOT gates are not native and require an ancilla qubit in order to implement. Furthermore, the synthesis of large ancilla patches containing GHZ states can be performed simultaneously as the preparation of a CNOT ancilla. Thus, CNOT and GHZ state injection layers have the same cost. A more general layer of nearest neighbor interactions, such as a layer of SWAP gates, may need as much time as three GHZ state injection layers. The situation is reversed in a model in which some two-qubit gates are native. In such an architecture, GHZ state preparation (see Fig. 4) and subsequent injection require three CNOT layers and additional measurement feedback. A similar argument applies to the parity check qubits: in a surface code architecture, the parity checks can be performed with no additional space cost, but this is not the case in a near-term architecture.

Finally, we establish some nomenclature for the variant architectures. All of the models considered in this article can be naturally implemented on a surface code architecture with lattice surgery as in [2]. Logical qubits are encoded into square surface code patches, which are arranged in a lattice. Nearest neighbor $X \otimes X$ or $Z \otimes Z$ measurements can be implemented through patch fusion. Through some ancilla padding, this allows the implementation of nearest neighbor CNOT gates and the preparation of long-range GHZ states on contiguous regions of logical qubits.

The LNN is a simple architecture featuring several data qubits in a line. We allow arbitrary single-qubit gates and arbitrary two-qubit gates on nearest neighbors along the line. A slightly more complex model is a "Linear GHZ Bus Architecture," where a line of qubits is connected to a parallel line of ancilla qubits [see Fig. 1(a)]. The ancilla qubits are reserved exclusively for the preparation of GHZ states. Once prepared, a GHZ state can be "injected" into the data qubits to implement long-range operations using circuits shown in Fig. 1(c) and (d). A surface code quantum computer also permits a 2-D "square lattice architecture" where data qubits are in a grid and two-qubit gates are possible on nearest neighbors (cf. [7]). Just like the linear GHZ bus, such a square lattice architecture may also be equipped with chains of ancilla qubits dedicated to GHZ state preparation. The "dual snake architecture," depicted in Fig. 1(b), adds two such intertwining chains of ancilla qubits, which may be used for the simultaneous preparation and injection of GHZ states.

In comparing the performance of these architectures, we measure cost as the minimum depth of the worst possible Clifford gate. In architectures with GHZ buses, we count the number of GHZ injections: implementation of a long-range gate through circuits in Fig. 1(c) and (d). Otherwise, we consider CNOT depth. We consider single-qubit gates free, due to their simple implementation as patch rotations.

B. PRIOR WORK

We briefly compare our approach to some other works. First, Devulapalli et al. [8] consider a broader family of connectivity graphs than LNN, but instead of GHZ states only focus on Bell state-enabled long-range swaps, and their impact on implementing permutation circuits. Our interest in LNN specifically is that it serves as a stepping stone toward square-lattice architectures that likely capture surface code constructions' capabilities on superconducting hardware. While the study of permutation circuits is a natural approach for quantifying the power of ancilla-enabled long-range gates, we find Clifford circuits enable a richer family of optimizations.

Second, Beverland et al. [7] investigate the performance of a square lattice architecture in which each data qubit is padded with three additional ancilla qubits for routing, leading to the total qubit footprint of $4n + o(n)$ for a program with n data qubits. In comparison, our dual snake architecture merely adds one ancilla per data qubit, with a total qubit footprint of $2n + o(n)$. Is the dual snake architecture slower for implementing Clifford gates than a layout with more ancillae? A theoretical analysis of the performance of the parallel CNOT routing considered by Beverland et al. [7] yields that $O(n^{1.5})$ layers suffice: n CNOT layers suffice to implement a Clifford gate, and each layer requires $\Theta(\sqrt{n})$ operations. This bound is much looser than the $2n + 1$ synthesis bound [11] with just a single entanglement bus, as well as the one we derive for our dual snake scheme, which achieves $\lceil \frac{3}{2}n \rceil + O(\sqrt{n})$, despite both needing fewer ancillae. We note that our dual snake architecture has GHZ buses that cross each other. This is permitted since Beverland et al. [7] show that two Bell states can still be prepared simultaneously in such a layout.

Third, Devulapalli et al. [8] consider implementing permutations of qubits using long-range swaps facilitated by long-range Bell state preparation. This work differs in many regards. We are interested in the capabilities of multiqubit interactions facilitated by GHZ state injections rather than two-qubit swaps. Also, they attempt to attain asymptotic speedups over routing based on nearest neighbor swaps, whereas we focus on the leading coefficient. This is because we consider worst case Clifford synthesis, which is already known to require $\Theta(n)$ depth in any of the models we consider, while they consider permutations specifically. Indeed, they prove an $O(\sqrt{n} \log n)$ upper bound on the ratio between swap-based and teleportation-based routing for general graphs, but for lattices this ratio must be $\Theta(1)$. This is consistent with our findings for general Clifford gates. Certainly, the $O(n^{1-\alpha})$-depth protocol for implementing the π_{rainbow} permutation they consider can be implemented directly on the linear GHZ bus and cannot be improved in our larger gate set since the bipartite entanglement across a GHZ state is the same as that of a Bell state.

Fourth, we note that the architecture admits a "clique flip" operation [defined in Fig. 5(a)], which is equivalent to applying CZ on all pairs of the k qubits it acts on. The name

IEEE Transactions on Quantum Engineering

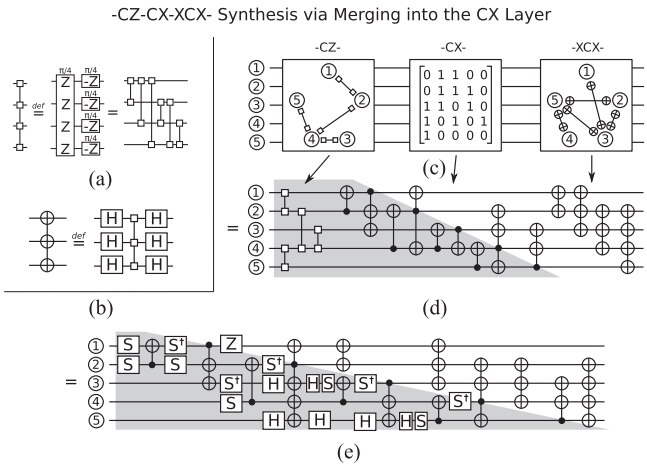

FIGURE 5. (a) "Clique flip" operation is a particular Pauli rotation with local corrections and can be shown to implement CZ on all pairs of the involved qubits. This operation is also considered by Maslov and Nam [10] and van de Wetering [11]. (b) XCX gates have an analog of the clique flip operation. (c) Clifford circuit is equivalent to -CZ-CX-XCX- up to some local gates. (d) CX circuit is synthesized using Proposition 6 with the controls on qubits with descending labels, and the -XCX- circuit can be built into an upward-facing triangle using Proposition 2. We do not care about the CZ synthesis since in (e) we use the method from Proposition 7 to absorb the CZ circuit into the downward-facing part of the CX circuit. Since XCX clique flips commute with CNOT fan-out targets, we commute them through and stack on top of the CNOTs in the odd layers.

of this operation is motivated by CZ circuit synthesis: CZ circuits are equivalent to graphs, and the clique flip operation lets us toggle all the edges of the graph within a clique of our choosing. This operation appeared in [10] and [11] as a special case of the global Molmer–Sorensen (GMS) gate. Since clique flip operations are local-Clifford-equivalent to Pauli rotations $\exp(i\frac{\pi}{4}P)$, we find that the constructions from this line of work already capture some, but not all, of the power of the model we consider. Indeed, other than the clique flip operation, GMS gates and GHZ injections seem to have rather different capabilities and resource requirements. On the one hand, recent work [12] shows that general GMS gates can implement Clifford gates in constant depth. On the other hand, GMS gates are inspired by Hamiltonian evolution on hardware with all-to-all connectivity (like ion trap quantum computers). Our GHZ bus model is more inspired by the limited connectivity of superconducting quantum computers running surface codes. While GHZ state injection can implement clique flips, a special case of the GMS gate, it is not clear how to use GHZ injection to implement general GMS gates. Similarly, while a unitary circuit with two Clique flips suffices to implement fan-out, it is unclear how to perform CNOT fan-out with just one clique flip.

Fifth, it is a well-known result in the theory of measurement-based quantum computation that Clifford gates can be implemented in constant depth on photonic hardware [15]. This is achieved by rendering the Clifford circuit into a sequence of gate teleportations, causing the overall *width* of the circuit to scale with the circuit complexity instead. We are interested in superconducting architectures where the width of the circuit is fixed.

Finally, we briefly discuss the feasibility of implementing this model in current-generation IBM hardware. Broadly, the hardware seems to have the necessary capabilities: mid-circuit measurement and feedforward correction via *dynamic circuits*, as well as connectivity that enables a limited version of the linear GHZ bus model. The GHZ injection circuits present the opportunity for significant savings in circuit depth, which may be useful when coherence time is a limitation. However, the mid-circuit measurements they require also introduce a lot of new noise, and qubits reserved for GHZ states cannot store data. Under what circumstances are these sacrifices worth the improvement in depth?

III. DEPTH-OPTIMIZED CLIFFORD SYNTHESIS USING GHZ STATES

First, recall that up to a layer of single-qubit Pauli gates, the group of Clifford operations on n qubits is isomorphic to the $2n \times 2n$ binary symplectic group $Sp(2n, \mathbb{F}_2)$. The task of synthesizing Clifford operations using a restricted set of gates is represented by the diagonalization of a binary symplectic matrix using operations corresponding to the gate set. For example, when we have only two-qubit entangling gates {CX, CZ}, Clifford operations can be decomposed into a layered computation in the form -L-CX-CZ-H-CZ-L- [2], [19], where -L- denotes a layer of single-qubit Clifford gates, -CX- and -CZ- denotes layers of circuits consisting entirely of CX and CZ gates, and -H- denotes a layer of Hadamard gates applied to all qubits.

Some Clifford operations can be synthesized with GHZ state injections with simpler circuits. For example, both fan-out gates and clique flip operations can be implemented using only one GHZ state injection, while they require a circuit of depth $\Omega(n)$ in LNN when only two-qubit gates are available. In general, Pllaha et al. [5] gave an algorithm that, with some exceptions, decomposes a Clifford operation into a minimal number of Pauli rotations $\exp(i\frac{\pi}{4}P)$. Their decomposition achieves a depth of $\leq 2n + 1$ that can be implemented naturally within our GHZ bus model. However, it is not often easy to obtain this decomposition since a subroutine it relies on—the triangularization of binary matrices by congruence—fails in some exceptional cases [20]. Furthermore, while Botha [20] does not give an explicit algorithm for obtaining these decompositions, we found that an algorithm based on their work requires $O(n^4)$ time. Finally, though the decomposition by Pllaha et al. [5] is guaranteed to use the minimal *number* of global operations, it does not leverage our models' additional powers, such as the ability to parallelize multiple GHZ state injections.

In the rest of this section, we will present constructive propositions for synthesizing various Clifford circuits, such as CZ circuits, CX circuits, and Hadamard-free circuits using GHZ state injections. Together, they lead to the two main results: first, in the linear GHZ bus model shown in Fig. 1(a), we show that any Clifford operation can be decomposed

into a circuit with at most $2n + 1$ GHZ state injection layers using $O(n^3)$ classical computation time. Second, we show that square lattice connectivity supporting the dual snake architecture in Fig. 1(b) can do so in depth $\leq \lceil \frac{3}{2}n \rceil + O(\sqrt{n})$. In addition, since graph states can be prepared using a CZ circuit, our constructions can also be extended to graph state synthesis. Unless otherwise specified, we will often use depth to refer to a circuit's GHZ state injection depth, that is, the number of parallel GHZ state injection stages.

A. -CZ- TRANSFORMATIONS AND GRAPH STATE SYNTHESIS

We first establish some results on synthesizing -CZ- layers. The central idea underpinning these methods is that -CZ- layers are equivalent to graphs since CZ gates are symmetric, self-inverse, and mutually commuting. A particularly convenient k-qubit gate for manipulating these graphs is the "clique flip" operation defined in Fig. 5(a), which implements a CZ gate on all pairs of qubits involved.

We represent an n-qubit CZ transformation as a graph $G(V, E)$, where each vertex in V corresponds to a qubit, and each edge $(v_1, v_2) \in E$ indicates a CZ gate between qubits v_1 and v_2. Since all CZ gates commute and are self-inverse, concatenating two CZ transformations $G_1(V, E_1)$, $G_2(V, E_2)$ gives a new CZ transformation $G_3(V, E_3)$, where E_3 is the symmetric difference of E_1 and E_2. With a slight abuse of notation, let us also denote G as the adjacency matrix. Then, $G_3 = G_1 \oplus G_2$. We assume without loss of generality (WLOG) that all CZ transformations share a common set of vertices. Note that in this representation, the application of a clique flip corresponds to the concatenation of a complete graph on a subset of vertices; in other words, it "flips" all the edges corresponding to a clique, hence the name.

By relating -CZ- circuits to graphs and their adjacency matrices, we can arrive at the following bound.

Proposition 1: Let $G(V, E)$ represent an n-qubit CZ circuit, and let $t(G)$ be the minimum number of clique flips required to implement G. Then, $\text{minrank}_2(G) \leq t(G) \leq \text{minrank}_2(G) + 1$, where

$$\text{minrank}_2(G) = \min\{\text{rank}_{\mathbb{F}_2}(D \oplus G) | D \in \text{diag}(\{0, 1\}^n)\}.$$

Proof: If $t(G) = 1$, then G contains exactly one clique, and $\text{minrank}(G) = t(G) = 1$. In particular, the minrank is achieved by choosing $D = I$.

First, let us show $\text{minrank}(G) \leq t(G)$. Suppose $t(G) = m$ for some $m > 1$, $G = K_1 \oplus \ldots \oplus K_m$ where $t(K_i) = 1$ for each $i \in [m]$. Since minrank is subadditive, $\text{minrank}_2(G) = \text{minrank}_2(\bigoplus_{i=1}^m K_i) \leq \sum_{i=1}^m \text{minrank}_2(K_i) = t(G)$.

Then, we'll show $t(G) \leq \text{minrank}(G) + 1$. Suppose $\text{minrank}_2(G) = r$; then, there exists $G^* = D^* \oplus G$ where $\text{rank}_{\mathbb{F}_2}(G^*) = r$. Since G^* is symmetric, we can use Lempel's factorization [21] to find an $n \times r'$ dimensional factor F such that $G^* = FF^T$, where $r' = r + 1$ if $G^* = G$ and $r' = r$ otherwise. Let f_i be the ith column of F, we can rewrite $G^* = \bigoplus_{i=1}^{r'} f_i f_i^T$. Note $f_i f_i^T = \text{diag}(f_i) \oplus K_i$, where K_i is a complete graph on vertices $\{j | F_{ij} = 1\}$. Therefore, $t(G) = t(K_1 \oplus \cdots \oplus K_{r'}) \leq \text{minrank}_2(G) + 1$. ■

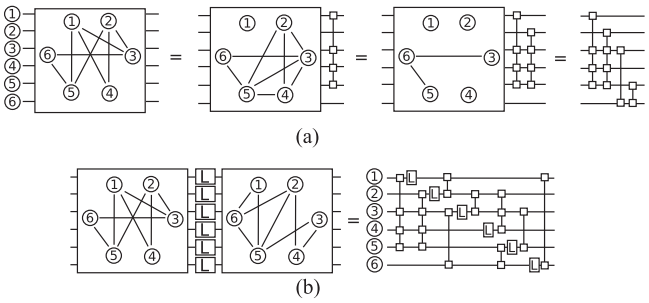

FIGURE 6. Synthesis of CZ circuits using the clique flip operation defined in Fig. 5(a). (a) Synthesis of a CZ circuit using $\leq n - 1$ clique flips from Proposition 2, also shown by van de Wetering [11]. (b) Illustration of the optimization from Proposition 3 with two examples of -CZ- layers given by the graphs in the figure. While each CZ layer individually can be synthesized using $\leq n - 1$ clique flips following Proposition 2, two such circuits can be slotted together to optimize depth.

We can show stronger bounds when considering specific classes of CZ circuits. One example application of Proposition 1 is on CZ circuits represented by random graphs. We can consider Erdős–Rényi random graphs $G(n, 1/2)$, where each edge appears with probability $1/2$. We know that for any $g \in G(n, 1/2)$, $\text{minrank}(g) > n - \sqrt{2n}$ almost always as $n \to \infty$ [22]. It also follows that the number of clique flips needed for a graph sampled randomly from $G(n, 1/2)$ is almost always about n once n is large enough. That is, for any $\epsilon, \delta > 0$, there exists an n^* s.t. for all $n \geq n^*$, $t(g) \geq (1 - \epsilon)n$ with probability $1 - \delta$ for randomly sampled $g \in G(n, 1/2)$.

Similar to the drawbacks of [5], computation and depth optimization of the circuit becomes a nontrivial task even though the algorithm guarantees the optimal number of clique flips needed. Hence, we will also survey a simpler method by van de Wetering [11], which works by disentangling qubits one by one, as illustrated in Fig. 6(a).

Proposition 2 (See [11]): Any CZ transformation can be implemented as a circuit using at most $n - 1$ GHZ state injections.

Proof: To synthesize $G(V, E)$, we can find $G_1, \ldots, G_m$ s.t. $G_1 \oplus \cdots \oplus G_m = G$, where each G_i consists of a clique implementable using one clique flip via a GHZ state injection. There is a simple algorithm to find these cliques. For each $i \in [n]$, let $S_i = [\bigoplus_{j=1}^{i-1} G_j] \oplus G$ be the graph left over after applying all G_j up to $i - 1$, and set G_i to be the complete graph on $N_{S_i}(v_i) \cup \{v_i\}$,[2] that is, a clique on v_i and its neighbors in S_i. Notice that v_i becomes an isolated vertex in $S_{i+1} = G_i \oplus S_i$ since the concatenation of G_i will cancel out any edges from v_i in S_i. It follows that $S_n = [\bigoplus_{i=1}^{n-1} G_i] \oplus G$ have only isolated vertices; hence, $G = G_1 \oplus \cdots \oplus G_{n-1}$, as desired. ■

van de Wetering et al.'s [11] construction illustrates a key optimization opportunity: "stacking." We observe that as

[2] $N_G(v)$ refers to the v's neighbors in G, i.e., $N_{G(V,E)}(v) := \{u | u \in V, (u, v) \in E\}$.

IEEE Transactions on Quantum Engineering

each clique flip disentangles a qubit, successive clique flips contain more isolated vertices, giving ample opportunities for parallelization. If we pick particular orders to disentangle the qubits, we can arrange the circuit in various "staircases." Using this idea, we arrive at the following constructions for parallelized circuits implementing -CZ- layers.

Proposition 3: -CZ-L-CZ- can be implemented as a circuit using $2n-2$ clique flips, implementable in GHZ-state-injection depth $n+1$ using a linear GHZ bus.

Proof: First, let the vertices be ordered from 1,... to n. For the first CZ transformation, pick $G^1 = G^1_1 \oplus \cdots \oplus G^1_{n-1}$ as in Proposition 2 where G^1_i accounts for the CZ gates related to v_i. Then, $v_1, \ldots, v_{i-1}$ are isolated vertices in G^1_i; hence, the corresponding clique flip does not act on qubits $1, \ldots, i-1$. This will be our first staircase.

For the second CZ transformation, let us fix the vertices in reverse: pick $G^2 = G^2_1 \oplus \cdots \oplus G^2_{n-1}$, where G^2_i accounts for the CZ gates related to v_{n-i+1}. Here, the clique flip corresponding to G^2_i only acts on $v_1, \ldots, v_{n-i}$. This will be our second upside-down staircase.

It follows that for $i = 3, \ldots, n-2$, G^1_i and G^2_{n-i+1} can be implemented in parallel, giving us a total depth of $n-3+4 = n+1$. ■

Finally, we give a construction that exploits the power of the dual snake model to implement two clique flips simultaneously even if their supports overlap, provided that they act on disjoint sets of qubits. This capability synthesizes a CZ transformation using GHZ-state-injection depth $\lceil n/2 \rceil + 1$.

Proposition 4: Any CZ transformation can be implemented as a circuit with GHZ-state-injection depth $\lceil n/2 \rceil + 1$ in an architecture with two parallel GHZ buses, such as the dual snake architecture.

Proof: Let us find a bipartition of the vertices $V = V_l \sqcup V_r$, where $V_l = \{v_1, \ldots, v_{\lceil n/2 \rceil}\}$ and $V_r = \{v_{\lceil n/2 \rceil + 1}, \ldots, v_n\}$. This bipartition defines a cut on G.

We will first address the CZ gates that cross the cut. For $i = 1, \ldots, \lceil n/2 \rceil$, let $S_i = [\bigoplus_{j=1}^{i-1} G^c_i] \oplus G$ and let C_i be the edges that cross the cut in S_i, where G^c_i is constructed as

1) two cliques, one on $V^l_i = N_{C_i}(v_i) \cup \{v_i\}$, and one on $V^r_i = N_{C_i}(v_{n-i+1}) \cup \{v_{n-i+1}\}$, if $(v_i, v_{n-i+1}) \notin C_i$;
2) one clique on vertices $V_i = V^l_i \cup V^r_i$, if $(v_i, v_{n-i+1}) \in C_i$.

In either cases, we observe that: 1) C_{i+1} does not contain any edges that have endpoints v_i, v_{n-i+1} and 2) vertices $v_1, \ldots, v_i$ and $v_n - i + 1, \ldots v_n$ are isolated in G^c_{i+1}. As a result of 1), $C_{\lceil n/2 \rceil + 1}$ is empty; it remains to deal with the edges contained in V_l and V_r. Given 2), we can implement the clique flips for the two disconnected subgraphs in parallel using the staircases given in Proposition 3. The parallelization increases the depth by at most 1. ■

The main idea for Proposition 3 is to disentangle qubits in opposite orders so they can be "stacked" together, as illustrated in Fig. 6(b), and the main idea for Proposition 4 is to cut the graph in two halves, disentangle across the cut, and deal with remaining edges in the two subgraphs in parallel, as illustrated in Fig. 7. Proposition 4 will later play a central role in our construction for general Clifford gates.

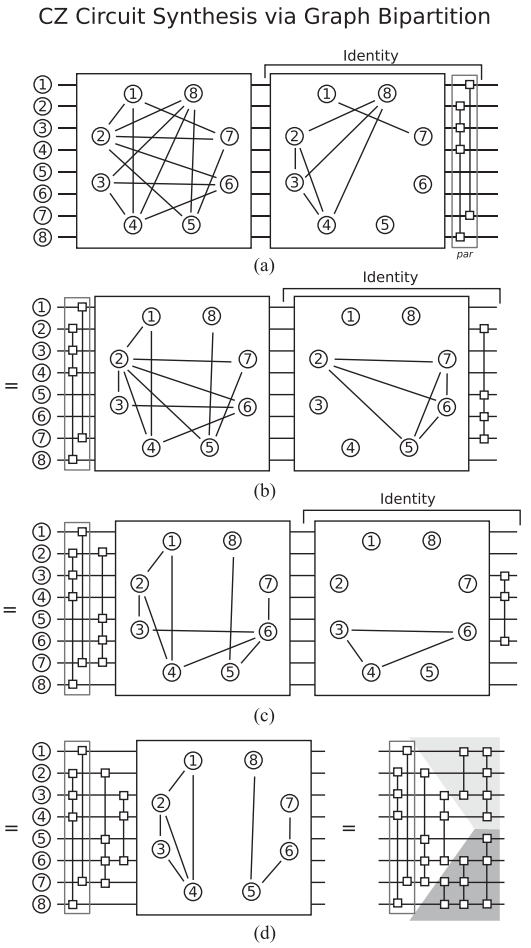

FIGURE 7. Example of the synthesis algorithm from Proposition 4. The algorithm first severs all edges between the groups $V_l = \{1, 2, 3, 4\}$ and $V_r = \{5, 6, 7, 8\}$. (a) Considering qubits 1 and 8 that are not connected (case 1), we can eliminate the edges across the cut using two clique flips. While these clique flips cannot be parallelized in a model with one GHZ bus, they can be with two GHZ buses. (b) Considering qubits 2 and 7 that are connected (case 2), we can eliminate the edges with one clique flip. (c) Similarly, qubits 3 and 6 correspond to case 2. We have removed all edges across the bipartition using $\lceil n/2 \rceil$ layers. (d) Finally, the two remaining graphs on V_l and V_r can be synthesized using Proposition 2, and due to the triangular structure of the circuits, this requires only one additional layer.

Aside from being useful for Clifford synthesis, the ability to implement arbitrary -CZ- transformations is also closely related to stabilizer state preparation. As shown in [23], all stabilizer states are equivalent to graph states up to an -L- layer, which are a -CZ- layer applied to $|+\rangle^{\otimes n}$. Thus, Proposition 4 shows that the dual snake architecture can also prepare stabilizer states in depth $\lceil n/2 \rceil + 1$.

B. -CX- TRANSFORMATIONS

Now, we turn our focus to -CX-, denoting an n-bit linear reversible function. Let us represent -CX- as an invertible binary matrix M, where each column of M corresponds to

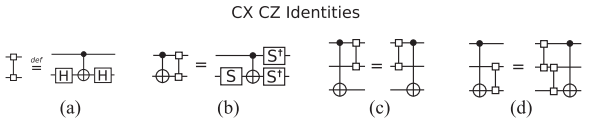

FIGURE 8. (a)–(d) Some identities for commuting a CZ gate through a CNOT gate.

the output state of a qubit in terms of the inputs. Traditionally, we find a circuit for -CX- by diagonalizing M with column operations (corresponding to CNOT gates) [1], which has the best-known depth of $5n$ in LNN [1] and $n + o(n)$ in all-to-all [4].[3] With a GHZ bus, we unlock additional abilities to perform up to n row operations simultaneously using fan-out and fan-in gates,[4] which are implementable using one GHZ state.

Proposition 5: Up to a relabeling of qubits, any linear reversible function can be implemented using n fan-outs using a linear GHZ bus.

Proof: For $i \in [n]$, let c_i be the ith column of M and let σ_i be the index of the first nonzero element of c_i. With one fan-out, we can add c_i (modulo 2) to all columns c_j, $i \neq j$, where $c_j[\sigma_i]$ is nonzero. This reduces M to the permutation matrix given by $[n] \mapsto \{\sigma_1, \ldots, \sigma_n\}$. ■

Proposition 6: Any linear reversible function can be implemented as a circuit with GHZ-state-injection depth $2n - 1$ using a linear GHZ bus.

Proof: To ensure the previous algorithm reduced M to a trivial permutation matrix, we need to ensure $\sigma_i = i$ for each i. This requires up to one long-range CNOT per fan-out: if $\sigma_i = i$, we are already done; otherwise, we can find c_j, where $\sigma_j = i$, and add c_j to c_i using one long-range CNOT. Such c_j always exists for some $j \geq i$; otherwise, M cannot be full rank.

We notice that after performing $n - 1$ fan-outs controlled by qubits $1, \ldots, n - 1$, the last column must have $n = \sigma_n$. Therefore, no additional CNOT is needed for the last fan-out, giving us the depth of $2n - 1$ as desired. ■

C. HADAMARD-FREE CLIFFORD TRANSFORMATIONS

Recall that in LNN, a -CZ- layer immediately adjacent to a -CX- layer can be implemented at no additional cost [3]. This fact also holds in the new model: we give a method for absorbing the -CZ- layer into the -CX- layer. The method below relies on circuit identities relating CZ and CX shown in Fig. 8. For additional clarity, we also give an example of the procedure in Fig. 9.

Proposition 7: Up to a permutation, any Hadamard-free Clifford transformation can be implemented as a circuit with n fan-outs using a linear GHZ bus.

Proof: We begin with the fact that a Hadamard-free Clifford transformation can be computed as a three-stage computation, -L-CX-CZ-. Let -CX- be written as n fan-outs

[3]The asymptotic optimal depth is $O(\frac{n}{\log n})$ [24]. We do not consider it here due to its impractically large constant overhead.

[4]A fan-in is equivalent to fan-out up conjugation by a layer of Hadamard gates.

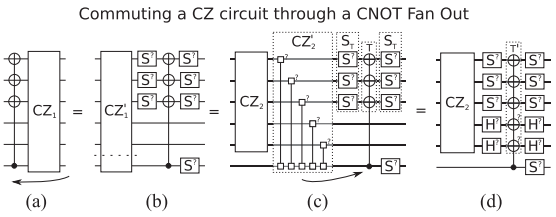

FIGURE 9. Example of the optimization performed in Proposition 7, which absorbs a CZ circuit into a sequence of CNOT fan-outs. (a) We leverage the identity in Fig. 8(b) to absorb some gates from CZ_1 into some S gates acting on $T \subset [n]$, resulting in CZ_1'. (b) Gates in CZ_1' touching the control qubit of the fan-out are extracted into CZ_2', with CZ_2 left over. (c) CZ_2' is absorbed into the fan-out by either removing some S gates or adding additional targets conjugated by H (resulting in $T' \subset [n]$). (d) After commuting, the CZ circuit does not touch the target of the fan-out anymore.

as in Proposition 5. First, as illustrated in Fig. 9, we can commute a layer of CZ gates through fan-out gates while reducing the width of the -CZ- layer. Given -CZ- circuit CZ_1 on qubits $1, \ldots, n$, and a fan-out gate F with control k and targets $T_F \subset [n] - \{k\}$, we will describe in three steps how this commutation is achieved.

1) Commute CZ_1 through F using well-known circuit identities given in Fig. 8. We have $F \cdot CZ_1 = CZ_1' \cdot S_T \cdot F \cdot S_{T \cup n}$, where S_T denotes a layer of single-qubit phase gates on qubits in T.
2) Partition $CZ_1' = CZ_2 \sqcup CZ_2'$, where $CZ_2 = \{\mathrm{CZ}(i, j) | \mathrm{CZ}(i, j) \in CZ_1', i, j \neq n\}$, and $CZ_2' = CZ_1' - CZ_2$. That is, CZ_2' consists of all CZ gates on qubit n and CZ_2 consists of all other CZ gates.
3) There are two cases for gates $\mathrm{CZ}(i, n) \in CZ_2'$.

a) $i \in T$: In this case, there exists $\mathrm{CX}(n, i) \in F$; the CZ gate can be implemented using phase gates;
b) $i \notin T$: Rewrite CZ (i, n) as H_i $\mathrm{CX}(n, i) H_i$; $\mathrm{CX}(n, i)$ can be merged with F and implemented with no additional cost. We obtain a new fan-out gate $F' = F \cup \{\mathrm{CX}(i, n)\}$, which is controlled by qubit n (same as F) with targets $T' = T \cup \{i\}$.

It follows that $F \cdot CZ_1 = CZ_2 \cdot F'$, where CZ_2 does not contain any CZ gates that act on the control of F, and F' is a fan-out gate with the same control as F and (possibly) more targets, up to conjugation by single-qubit phase gates and Hadamard gates. F and F' can both be implemented using one GHZ state injection, up to some irrelevant single-qubit gates.

We can repeatedly commute the CZ circuit to obtain $CZ_2, \ldots, CZ_n$. Since the n fan-out gates given by Proposition 5 have distinct controls, each time we pass by a fan-out layer, the width of the CZ circuit decreases by 1. Hence, $CZ_n = I$; we have implemented -CZ- inside -CX- with no additional cost, as desired. ■

A simple corollary follows that a Hadamard-free operation can be implemented in depth $2n - 1$, since we can still easily commute CZ gates through the additional CNOT layers. Furthermore, we also notice that if CZ_1 does not act on qubits

$Q_n = \{i_1, \ldots, i_m\}$, and $k \notin Q_n$, where k is the control of F, then CZ_2 does not act on Q_n, and F' does not add additional targets to qubits in Q_n. For all intents and purposes, the commutation rule given above leaves gates on Q_n unchanged.

Given this observation, in fact, a second -CZ- layer can also be implemented at no additional cost when implementing a -CX- circuit exactly using a GHZ bus. We will show this construction in the next subsection.

D. CLIFFORD TRANSFORMATIONS

Putting everything together, we arrive at the main result for the linear GHZ bus model. An example of this algorithm is given in Fig. 5.

Corollary 1: Any Clifford transformation can be implemented as a circuit with GHZ-state-injection depth $2n + 1$ using a linear GHZ bus.

Proof: Let us first consider an alternative decomposition of Clifford operations, -L-CZ-CX-XCX-L-, where -XCX- denotes a layer of X-controlled-NOT-gates: $\mathrm{XCX} := H^{\otimes 2} \cdot \mathrm{CZ} \cdot H^{\otimes 2}$. We can obtain this decomposition by commuting the full layer of Hadamard gates through the second -CZ- layer in the scheme given by Bravyi and Maslov [19].

First, we can synthesize the -CX- layer using Proposition 6, where odd layers $2i - 1$ contain one CNOT gate on qubits i, j, where $j > i$, and even layers $2i$ contain a fan-out controlled by qubit i. From the left, we can push in a -CZ- circuit using techniques described in Proposition 7. From the right, we can first decompose the -XCX- circuit as an upside-down staircase using techniques described in Proposition 3, and commute them to stack on top of the CNOTs in the odd layers. This is always possible since XCX gates commute with the target of a CNOT gate, and the controls of the fan-outs are in descending order. Overall, the depth is increased by 2. ■

We also present a similar result in the dual snake model.

Corollary 2: Any Clifford transformation can be implemented as a circuit with GHZ-state-injection depth $\lceil \frac{3}{2} \rceil n + O(\sqrt{n})$ in a square-lattice architecture supporting the dual snake layout.

Proof: It is sufficient to be able to implement a Hadamard-free transformation and a CZ transformation [19]. Up to a permutation, a Hadamard-free Clifford transformation can be implemented in depth n by Proposition 7, and a -CZ- circuit can be implemented in depth $\lceil \frac{1}{2} \rceil n + O(1)$ by Proposition 4. Finally, a permutation can be implemented in depth $O(\sqrt{n})$ on a square lattice [25], where adjacent horizontal and vertical SWAPs can be implemented efficiently by using the ancilla qubits otherwise dedicated to the GHZ bus. ■

APPENDIX
APPENDIX LOWER BOUNDS FOR CLIFFORD CIRCUITS

Here, we present some simple counting arguments.

Proposition 8: Any sequence of m many n-qubit Pauli rotations that implements an arbitrary element of the Clifford group will require $m \geq n$.

Proof: Recall that $\log_2 |\mathcal{C}_n| \geq 2n^2 + n$ bits are required to specify an element of the Clifford group [19]. Each Pauli rotation on n qubits encodes $2n + 1$ bits, so at least $(2n^2 + n)/(2n + 1) = n$ are required. ■

More generally, since we allow Pauli rotations, fan-out gates following a GHZ state injection, as well as arbitrary single-qubit Clifford gates, we need to be more careful when deriving general lower bounds for our model.

Proposition 9: Any circuit consisting m layers of parallelizable Pauli rotation and fan-out gates that implements an arbitrary element of the Clifford group will require $m \geq 0.648n - 2$.

Proof: First, we observe that since the Pauli matrices are normalized by the Clifford group, we can commute all single-qubit Clifford gates to the beginning of the circuit. This may alter the elements of the Pauli rotation or change the controls and the targets of the fan-out gate to arbitrary Paulis (instead of the Z- and X-targets). Let us call them conjugated fan-out gates. In addition, since $\pi/2$ Pauli rotations are local, we may ignore the sign of a Pauli rotation. WLOG, we can consider a canonical form where circuits consist of one layer of single-qubit Clifford gates followed by m layers of parallelizable Pauli rotations and conjugated fan-out gates, and we are interested in finding a lower bound on m such that any n-qubit Clifford operation requires at least m such layers. Similarly as above, we proceed by finding an upper bound on the number of bits required to specify one such layer of global gates. We will furthermore assume that all the gates are conjugated fan-out gates since they require strictly more information to specify compared to a Pauli rotation acting on the same qubits.

Suppose that the layer of the global gate acts nontrivially on k qubits. Then, there are 3^k ways to specify the nonidentity matrices and $k - 1$ locations where the string can be uniquely split. It remains to specify a control qubit for each segment of the chain, for which there are at most l choices for a segment containing l nonidentity elements. Overall, with s segments, there are $l_1 \times \cdots \times l_s$ choices where $l_1 + \cdots + l_s = k$, which is upper-bounded by $2^{\frac{k}{2}}$ when $s = k/2$. The total number of possibilities is at most

$$\sum_{k=0}^{n} \left(3^k \cdot 2^{k-1} \cdot 2^{\frac{k}{2}}\right) = \frac{\left(6\sqrt{2}\right)^{n+1} - 1}{6\sqrt{2} - 1} \leq \frac{6\sqrt{2}}{6\sqrt{2} - 1} \left(6\sqrt{2}\right)^n. \tag{1}$$

Finally, there are 24^n choices for the layer of single-qubit gates. Since there are 2^{2n^2+n} Clifford operations, we will require at least

$$m \geq \frac{2n^2 + n - n\log(24)}{n\log\left(6\sqrt{2}\right) - \log\left(\frac{6\sqrt{2}}{6\sqrt{2}-1}\right)} \geq 0.648n - 2. \tag{2}$$

■

ACKNOWLEDGMENT

The authors thank Sergei Bravyi, Shelly Garion, Alexander Ivrii, Dmitri Maslov, and Derek Wang for helpful discussions.

REFERENCES

[1] S. A. Kutin, D. P. Moulton, and L. M. Smithline, "Computation at a distance," 2007, *arXiv:quant-ph/0701194*, doi: 10.48550/arXiv.quant-ph/0701194.

[2] D. Litinski, "A game of surface codes: Large-scale quantum computing with lattice surgery," *Quantum*, vol. 3, 2019, Art. no. 128, doi: 10.22331/q-2019-03-05-128.

[3] D. Maslov and W. Yang, "CNOT circuits need little help to implement arbitrary Hadamard-free clifford transformations they generate," *NPJ Quantum Inf.*, vol. 9, 2023, Art. no. 96, doi: 10.1038/s41534-023-00760-2.

[4] D. Maslov and B. Zindorf, "Depth optimization of CZ, CNOT, and clifford circuits," *IEEE Trans. Quantum Eng.*, vol. 3, 2022, Art. no. 2500408, doi: 10.1109/TQE.2022.3180900.

[5] T. Pllaha, K. Volanto, and O. Tirkkonen, "Decomposition of Clifford gates," in *Proc. IEEE Glob. Commun. Conf.*, 2021, pp. 1–6, doi: 10.1109/GLOBECOM46510.2021.

[6] D. Litinski and F. von Oppen, "Braiding by Majorana tracking and long-range CNOT gates with color codes," *Phys. Rev. B*, vol. 96, Nov. 2017, Art. no. 205413, doi: 10.1103/PhysRevB.96.205413.

[7] M. Beverland, V. Kliuchnikov, and E. Schoute, "Surface code compilation via edge-disjoint paths," *PRX Quantum*, vol. 3, May 2022, Art. no. 0020342, doi: 10.1103/PRXQuantum.3.020342.

[8] D. Devulapalli, E. Schoute, A. Bapat, A. M. Childs, and A. V. Gorshkov, "Quantum routing with teleportation," 2022, *arXiv:2204.04185*, doi: 10.48550/arXiv.2204.04185.

[9] A. Bapat, A. M. Childs, A. V. Gorshkov, and E. Schoute, "Advantages and limitations of quantum routing," *PRX Quantum*, vol. 4, Feb. 2023, Art. no. 010313, doi: 10.1103/PRXQuantum.4.010313.

[10] D. Maslov and Y. Nam, "Use of global interactions in efficient quantum circuit constructions," *New J. Phys.*, vol. 20, no. 3, Mar. 2018, Art. no. 033018, doi: 10.1088/1367-2630/aaa398.

[11] J. van de Wetering, "Constructing quantum circuits with global gates," *New J. Phys.*, vol. 23, no. 4, Apr. 2021, Art. no. 043015, doi: 10.1088/1367-2630/abf1b3.

[12] S. Bravyi, D. Maslov, and Y. Nam, "Constant-cost implementations of clifford operations and multiply-controlled gates using global interactions," *Phys. Rev. Lett.*, vol. 129, Nov. 2022, Art. no. 230501, doi: 10.1103/PhysRevLett.129.230501.

[13] L. Piroli, G. Styliaris, and J. I. Cirac, "Quantum circuits assisted by local operations and classical communication: Transformations and phases of matter," *Phys. Rev. Lett.*, vol. 127, Nov. 2021, Art. no. 220503, doi: 10.1103/PhysRevLett.127.220503.

[14] K. C. Smith, E. Crane, N. Wiebe, and S. M. Girvin, "Deterministic constant-depth preparation of the AKLT state on a quantum processor using fusion measurements," *PRX Quantum*, vol. 4, 2023, Art. no. 020315, doi: 10.1103/PRXQuantum.4.020315.

[15] R. Jozsa, "An introduction to measurement based quantum computation," 2005, *arXiv:quant-ph/0508124*, doi: 10.48550/arXiv.quant-ph/0508124.

[16] B. Coecke and R. Duncan, "Interacting quantum observables: Categorical algebra and diagrammatics," *New J. Phys.*, vol. 13, no. 4, Apr. 2011, Art. no. 043016, doi: 10.1088/1367-2630/13/4/043016.

[17] S. Debnath, N. M. Linke, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe, "Demonstration of a small programmable quantum computer with atomic Qubits," *Nature*, vol. 536, no. 7614, pp. 63–66, Aug. 2016, doi: 10.1038/nature18648.

[18] D. Litinski and F. von Oppen, "Lattice surgery with a twist: Simplifying Clifford gates of surface codes," *Quantum*, vol. 2, 2023, Art. no. 62, doi: 10.22331/q-2018-05-04-62.

[19] S. Bravyi and D. Maslov, "Hadamard-free circuits expose the structure of the Clifford group," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4546–4563, Jul. 2021, doi: 10.1109/TIT.2021.3081415.

[20] J. D. Botha, "Triangularizing matrices over GF(2) by congruence," *Linear Multilinear Algebra*, vol. 42, no. 2, pp. 109–158, 1997, doi: 10.1080/03081089708818495.

[21] G. Seroussi and A. Lempel, "Factorization of symmetric matrices and trace-orthogonal bases in finite fields," *SIAM J. Comput.*, vol. 9, no. 4, pp. 758–767, 1980, doi: 10.1137/0209059.

[22] S. Friedland and R. Loewy, "On the minimum rank of a graph over finite fields," 2010, *arXiv:1006.0770*, doi: 10.48550/arXiv.1006.0770.

[23] B. Zeng, H. Chung, A. W. Cross, and I. L. Chuang, "Local unitary versus local Clifford equivalence of stabilizer and graph states," *Phys. Rev. A*, vol. 75, Mar. 2007, Art. no. 032325, doi: 10.1103/PhysRevA.75.032325.

[24] J. Jiang et al., "Optimal space-depth trade-off of CNOT circuits in quantum logic synthesis," in *Proc. 14th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2010, pp. 213–229, doi: 10.1137/1.9781611975994.13.

[25] C P. Schnorr and A. Shamir, "An optimal sorting algorithm for mesh connected computers," in *Proc. 18th Annu. ACM Symp. Theory Comput.*, 1986, pp. 255–263, doi: 10.1145/12130.12156.