Universal Graph Theory Operations for Graph State Preparation

Tristan Cam, Cyril Gavoille, Yvan Le Borgne, Simon Martiel

Universal Graph Theory Operations for Graph State Preparation

Tristan Cam*
LaBRI, University of Bordeaux, CNRS
France

Cyril Gavoille
LaBRI, University of Bordeaux, CNRS
France

Yvan Le Borgne
LaBRI, University of Bordeaux, CNRS
France

Simon Martiel
IBM Quantum, IBM France Lab
France

Abstract

In quantum computing, graph states play a crucial role in quantum error correction and measurement-based quantum computing. Preparing these states efficiently on hardware with constrained connectivity is a fundamental challenge. In this work, we establish a universal framework for graph state preparation using only Controlled-Z (CZ) gates along the edges of a given hardware connectivity graph and local complementation operations. We prove that any graph state can be prepared using only these operations, providing a constructive transpilation method that transforms the input circuit into an equivalent one without increasing the number of entangling gates. Additionally, as our approach preserves entangling count and depth of the input circuit, we show that this framework also allows for optimal graph state preparation.

Keywords: Quantum circuit - Graph state - Transpilation.

1 Introduction

Quantum computing holds the potential to deliver exponential speedups over classical computing [JL03, PL23, Sho97, Vid03], with several demonstrations of such algorithmic advantages using ion trap [FML+17, WBD+19], superconducting [HBC+22, RHK+20], photonic [XLZZ21, ZCL+22], and Rydberg atom [EKC+22] quantum processors. However, achieving a universal fault-tolerant quantum computer capable of efficiently solving problems like large integer factorization would require millions of qubits with low error rates and long coherence times. Given the current limitations of noisy intermediate-scale quantum (NISQ) computers [CCL23, Pre18], research is focused on overcoming these device limitations, making the design of efficient quantum circuits crucial. Extensive research has been conducted on quantum circuit synthesis [AMMR13, DN06, MMN+16], particularly using restricted gate sets such as CNOT-, CZ- or Clifford-circuit synthesis [GdBM24, KdG20, KM13, MR18] with the goal of minimizing the number of entangling gates or the circuit depth.

*This work has been supported by the French ANR project Plan France 2030 (ANR-22-PETQ-0007), and also ANRT Program.

A key class of quantum states, known as stabilizer states, are particularly significant due to their properties in quantum error correction [FL11, KKL19]. In this work, we focus on graph states, a subclass of stabilizer states that can be represented by undirected graphs and that have been shown to be closely related to the structure of Clifford isometries [GdBMV25]. Van den Nest *et al.* [VdNDDM04] showed that any stabilizer state can be transformed into a graph state using single qubit Clifford gates, making the preparation of graph states equivalent to that of stabilizer states - a topic that has been widely explored [CDLTP11, DW18, KMY25]. Moreover, all local-Clifford equivalent graph states are related through repeated local complementation, a well-studied [AMSDS20, VdNDDM04] graph operation introduced by Kotzig [Kot68]. Another challenge in compilation problems is ensuring circuit compliance with hardware constraints. Quantum chip architectures do not always support all-to-all connectivity, and in practice, one often needs to work with only a connected subgraph of the physical architecture that offers sufficiently high fidelity, as error rates across qubit connections may be degraded. This motivates research on circuit compilation on arbitrary hardware.

Our contribution. Given a n-qubit hardware architecture corresponding to a connected graph H on n vertices and a graph G on at most n vertices, we show that there exists a circuit that prepares the graph state $|G\rangle$ by only toggling edges in H (corresponding to applying a CZ gate) and applying local complementations (corresponding to local Clifford gates on sets of neighboring qubits).

Our method relies on constructively transforming any H-compliant graph state preparation circuit C into a sequence of edge toggling in H and local complementation. Moreover, we show that the number of edge toggling in this sequence is bounded by the number of entangling gates in C, since our transpilation either keeps or deletes CZ gates of the original circuit. This gives a direct bound on both the two-qubit-gate count and depth in our output circuit. In particular, this entails that given an optimal algorithm to construct a Clifford circuit C that prepares a given graph state on H, our method gives a circuit that has a CZ-count and depth bounded by that of C and prepares the same state.

Remarkably, our result can also be expressed in the language of graph theory: given a connected graph H on n vertices, any graph G on n vertices can be constructed from the empty graph on n vertices with only local complementations and toggling edges of H. To the best of our knowledge, this result was previously unknown. We raise the question of finding a purely combinatorial proof.

Example. The following is a short example of our method. We give ourselves a circuit that prepares the complete graph state in CZ depth 3 without connectivity constraint. Here, we suppose that our hardware graph is a path, thus we rewrite the circuit with a SWAP gate, translate it to CZ and Hadamard gates and obtain a hardware-compliant circuit with a CZ depth of 6 (cf. Fig. 1).

Given such a hardware-compliant Clifford circuit that prepares a graph state, our method gives a explicit construction of the following equivalent circuit that only contains CZ gates and local Clifford gates arranged in local complementation layers, while preserving CZ depth and count. This allows the circuit to be read as a sequence of instructions to construct the desired graph using edge toggling and local complementation, as demonstrated in Fig. 2.

In this example, the CZ depth was reduced to 5, using the Vanishing CZ lemma defined

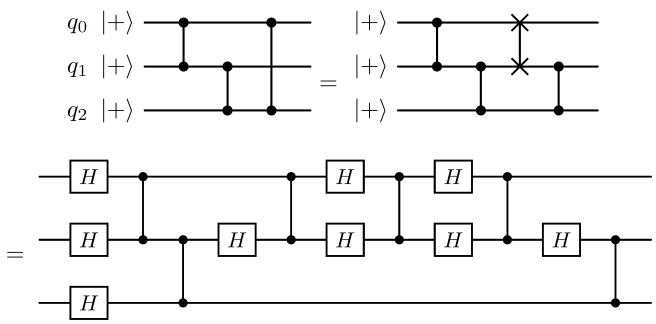

Figure 1: Circuit preparing the 3-qubit complete graph state naively transpiled from all-to-all to LNN architecture.

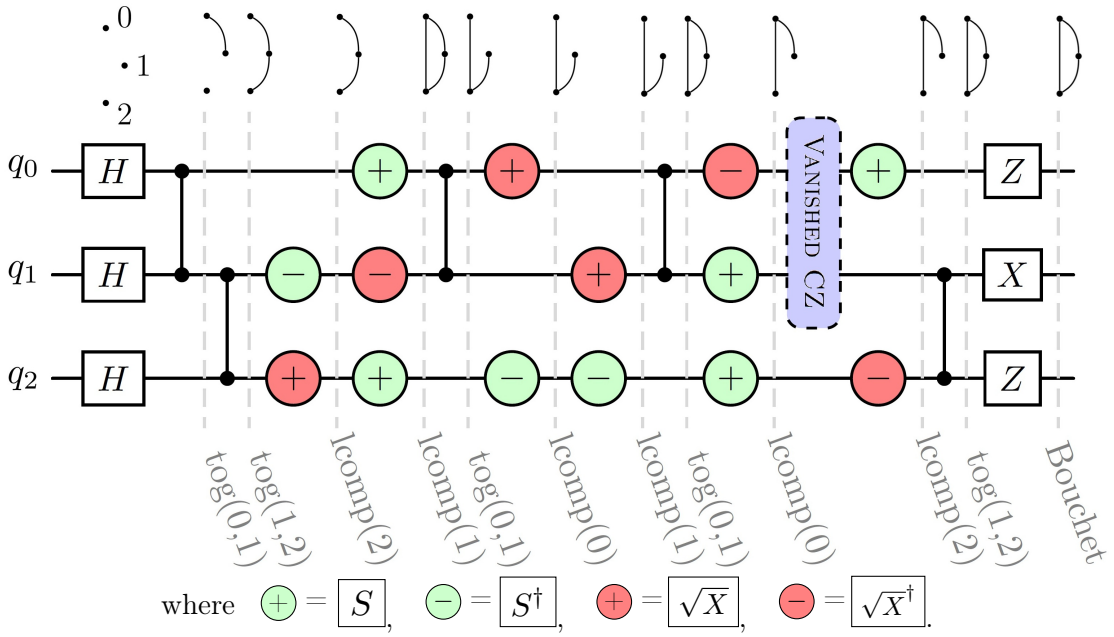

Figure 2: Equivalent circuit transpiled using our method, with corresponding graph instructions. The last layer of Clifford gates stabilizes this graph state (equivalently it can be seen as two local complementations on q_1), this can be computed using Bouchet's algorithm [Bou91].

later. Notice that our method did not produce the optimal circuit, as it would require only two CZ gates and one local complementation, which is minimal for any connected architecture.

2 Preliminary notions

A graph state $|G\rangle$ on n qubits is a stabilizer state represented by an undirected graph G. To prepare $|G\rangle$, one only has to prepare the state $|+\rangle^{\otimes n} = |\overline{K_n}\rangle$ and for each edge $\{u, v\} \in E(G)$ apply a controlled-Z (CZ) operation on the corresponding qubits u and v. This second step of the process can be done in any order as CZ gates commute. This method allows to prepare $|G\rangle$ using exactly $|E(G)|$ two-qubit gates, given all-to-all connectivity. Our aim is to replace some

two-qubits gate by emulating them with local Clifford gates acting as local complementations, while transpiling to any architecture.

2.1 Local complementation

Van den Nest *et al.* [VdNDDM04] showed that two graph states $|G\rangle$ and $|G'\rangle$ can be transformed into each other by applying only local Clifford gates if and only if the graphs G and G' are *locally equivalent* (denoted by $G \sim_{\text{loc}} G'$), i.e., there is a finite sequence of *local complementation* operations on G that results into G'. They also show that any of these sequences is feasible using a bounded number of local Clifford operations. This graph transformation is particularly convenient to reduce the number of edges of a graph (cf. Fig. 3). For instance, applying any local complementation on the complete graph on n vertices K_n results in the star graph $K_{1,n-1}$, hence $K_n \sim_{\text{loc}} K_{1,n-1}$. This means that instead of naively preparing $|K_n\rangle$ using $\Theta(n^2)$ CZ gates, one only has to prepare $|K_{1,n-1}\rangle$ using $\mathcal{O}(n)$ CZ gates then apply the local Clifford operators corresponding to a local complementation [DKPvdW20]. As reducing the number of error-prone two-qubits gates is crucial, this shows that allowing local complementation in addition to applying CZ gates along the edges of a hardware graph enables significant optimizations in graph state preparation. Let us define the simple graph operation of local complementation and rephrase it as quantum state identity.

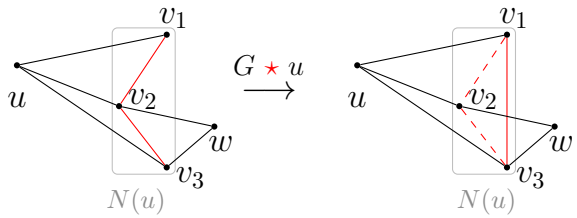

Figure 3: Local complementation of a graph G according to u. v_1 and v_3 are connected in $G \star u$ since the edge $v_1v_3 \notin E(G)$, while $v_1v_2 \in E(G)$ and $v_2v_3 \in E(G)$.

Definition 1 (Local complementation) *Let G be a graph and let u be a vertex of G. The* local complementation *of G according to u, written $G \star u$, is a graph which has the same vertices as G, but all the neighbors $v_1, v_2, \ldots$ of u are connected in $G \star u$ if and only if they are not connected in G. All other edges are unchanged.*

We will use the graph states statement of local complementation from [DP09]. Let us denote the rotation $R_x(-\pi/2)$ by the $\sqrt{X}^\dagger$ gate and the rotation $R_z(\pi/2)$ by the S gate.

Lemma 1 (Van den Nest [VdNDDM04]) *Let $|G\rangle$ be a graph state, u a vertex of G and $N(u)$ the set of neighbors of u in G. Then,*

$$\sqrt{X}^{\dagger(u)} S^{(N(u))} |G\rangle \;=\; \sqrt{X}^{(u)} S^{\dagger(N(u))} |G\rangle \;=\; |G \star u\rangle \;.$$

where $U^{(S)}$ is applying gate U on the set of qubits S.

Also, note that $G \star u \star u = G$. This translates to the following identity.

Proposition 1 (Local complementation, circuit) *Let $|G\rangle$ be a graph state, u a vertex of G and $N(u)$ the set of neighbors of u in G. Performing a* local complementation *on $|G\rangle$ according to u is done by:*

- *adding a $\sqrt{X}^\dagger$ (resp. $\sqrt{X}$) gate on the qubit u,*
- *adding a S (resp. $S^\dagger$) gate on every qubit corresponding to a vertex in $N(u)$.*

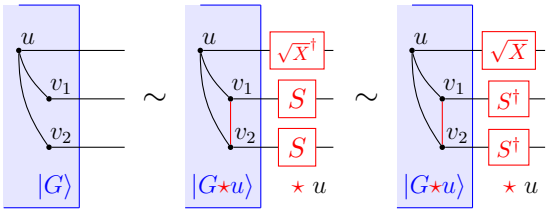

Figure 4: The three states above are equivalent: these identical local complementations expressed in three different ways can be created from the identity.

We will use this identity throughout our construction to repeatedly reveal two carefully placed identical local complementations on some set of qubits corresponding to $\{u, N(u)\}$ (using the fact that $G = G \star u \star u$) by adding local Clifford gates to these qubits in the circuit that prepares $|G\rangle$, and keeping track of their actions within the graph G itself.

2.2 Graph state preparation on arbitrary hardware

A number of different techniques [WBLW23, YZ24, ZHQ+21] can be used to *transpile* some given state preparation circuit into an hardware compatible circuit, and more recently by using discrete constraint optimization [BPBB25]. The most standard approaches rely on inserting SWAP gates to *route* qubits along edges of H in order to bring them closer and enable their interaction. The following proposition simply assumes that we are given such a transpilation algorithm, and does not require any structure on this algorithm. Since graph state can be prepared using CZ and Hadamard gates, and since every SWAP can be decomposed using three CZ gates and Hadamard gates, it is easy to see that the following proposition holds.

Proposition 2 *Let H be a connected graph. Suppose we have a quantum hardware whose connectivity between qubits is exactly described by H. Then any Clifford circuit can be implemented with local Clifford gates and CZ gates only on edges of H.*

Using this proposition to produce any H-compliant Clifford circuit C that prepares a certain graph state $|G\rangle$, we are able to transpile C into an equivalent circuit C' that obeys the following theorem.

Theorem 1 *Let H be a connected graph on n vertices. Let G be any graph on n vertices. The state $|G\rangle$ can be prepared by a circuit consisting of CZ gates along edges of H and local complementations.*

This means that – if we do not have a hard constraint on depth – any hardware can indistinguishably implement a circuit preparing any graph state using only our two operations.

3 Our construction

Informally, our iterative transpilation method will consist in essentially the same process repeated for each CZ gate of the input circuit to yield an equivalent circuit constructed with only CZ along the edges of the hardware graph H and local complementations. Crucially, each step either keeps or deletes the current CZ. Each step roughly consists of:

- first, extracting some generic local Clifford gates and a CZ gate from the *to-be-processed* subcircuit (represented on the right hand side of the figures);
- then, finding a way to weakly commute these local Clifford gates with the CZ, making heavy use of the local complementation circuit identity;
- finally, absorbing the CZ gate in the *processed* subcircuit (represented on the left hand side of the figures) that prepares the target graph state; and
- pulling all the single-qubit gates that cannot be interpreted as local complementations back into the *to-be-processed* subcircuit previously mentioned.

First of all, we introduce two handy quantum state identities in Lemma 2 and Lemma 3 that we will use in our construction. Their purpose is to commute specific local Clifford gates to the other side of a CZ gate (or in the case of Lemma 2, get rid of the CZ entirely) in some well-defined subcircuit. The proofs of Lemma 2 and Lemma 3 have been moved in Appendix A and Appendix B respectively.

In the setting of Lemma 2, we consider a CZ gate on qubits v_1, v_2 and the subcircuit preceding the CZ that prepares a graph state G_i, where v_1 has no other neighbor than v_2 in G_i and v_2 has at least one other neighbor in G_i. Then in this setting:

Lemma 2 (Vanishing CZ) *The two states depicted in Fig. 5 are equivalent.*

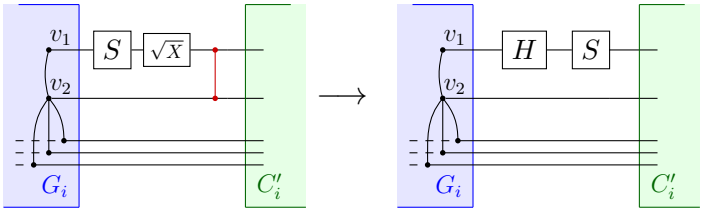

Figure 5: Vanishing CZ.

In the setting of Lemma 3, we consider a CZ gate on qubits v_1, v_2 and the subcircuit preceding the CZ that prepares a graph state G_i, where v_1 has at least one neighbor u that is not v_2 in G_i and v_2 may or may not be already connected to v_1 in G_i. Then in this setting:

Lemma 3 (Weak CZ commutation) *The two states depicted in Fig. 6 are equivalent.*

We are now able to prove to prove Theorem 1 by induction.

Proof of Theorem 1

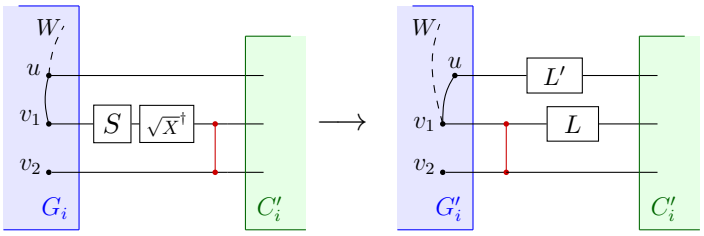

Figure 6: Weak CZ commutation, where $W = N_u \setminus \{v_1\}$, and L, L' are generic local Clifford gates.

Given H a connected graph of n vertices, let G be any graph with n vertices.

We define $\mathcal{C}_H$ as the set of all Clifford circuits where the qubits correspond to vertices of H, containing only local Clifford gates and CZ gates along edges of H. Moreover, let us define $\mathcal{C}_{H_{\text{loc}}}$ as the subset of $\mathcal{C}_H$ where all local Clifford gates correspond to local complementations (see Proposition 1). In the following, we will say that a graph G is *locally preparable* on H if G can be constructed from the empty graph on n vertices using only local complementation and toggling edges of H. Note that a sequence of instructions to locally prepare G on H directly translates to a circuit in $\mathcal{C}_{H_{\text{loc}}}$ that prepares $|G\rangle$.

Let C be any circuit given by Proposition 2 such that $C|0\rangle = |G\rangle$ and $C \in \mathcal{C}_H$. We will now show that there exists some circuit C' such that $C'|\overline{K_n}\rangle = C|0\rangle$ and $C' \in \mathcal{C}_{H_{\text{loc}}}$.

We will proceed by induction on the number of CZ in C. Let the proposition $P(i)$ for $0 \leq i \leq m$ where m is the number of CZ in circuit C be the following: there exists a graph G_i locally preparable on H and $C_i \in \mathcal{C}_H$ such that $C_i|G_i\rangle = C|0\rangle$ and C_i only contains $m-i$ CZ.

$P(0)$ is correct: pick $G_0 = \overline{K_n}$ which is trivially locally preparable on H and at the start $C_0 = CH^{\otimes n}$ (if C contains no CZ, then trivially the result of C is the same on any hardware with n qubits).

Let us now show that $P(i) \implies P(i+1)$. We now have $C_i|G_i\rangle = C|0\rangle$ and we would like to build G_{i+1} locally preparable on H and $C_{i+1} \in \mathcal{C}_H$ where C_{i+1} has $m-i-1$ CZ such that $C_{i+1}|G_{i+1}\rangle = C|0\rangle$.

Consider any first CZ appearing in the *to-be-processed* subcircuit C_i. In the general case this CZ gate on v_1, v_2 two qubits of $|G_i\rangle$ is preceded by two local Clifford gates L_1, L_2 (possibly being the identity) acting on v_1, v_2. Let us denote by C'_i the circuit obtained by removing L_1, L_2 and the current CZ gate from C_i (cf. Fig. 7).

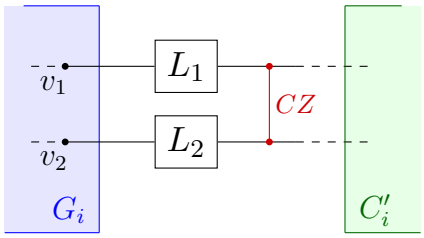

Figure 7: Simplified sketch of the i-th step. All other qubits are not shown.

Using Euler decomposition [MV06], we can find $a_1, b_1, c_1, a_2, b_2, c_2 \in \mathbb{Z}/4\mathbb{Z}$ such that $L_1 =$

$S^{c_1}\sqrt{X}^{\dagger b_1}S^{a_1}$, $L_2 = S^{c_2}\sqrt{X}^{\dagger b_2}S^{a_2}$. Notice that the first S gates in this expression can be commuted through the CZ and be absorbed back in the *to-be-processed* subcircuit C_i'. Thus, without loss of generality, we can assume that $L_1 := \sqrt{X}^{\dagger b_1}S^{a_1}$, $L_2 := \sqrt{X}^{\dagger b_2}S^{a_2}$.

Thus for b_1, b_2 the powers of $\sqrt{X}^\dagger$ in the decompositions of L_1, L_2, we now have three cases:

1. Both b_1 and b_2 are even.
2. Only one of b_1 and b_2 is odd.
3. Both b_1 and b_2 are odd.

We will show case-by-case that $P(i+1)$ holds.

Case 1. Both b_1 and b_2 are even, then $\sqrt{X}^{\dagger b_1} = X^{-b_1/2}$ and $\sqrt{X}^{\dagger b_2} = X^{-b_2/2}$ are both Pauli operators and can be weakly commuted through the CZ gate, inducing another Pauli operator on the *to-be-processed* side. Then the last gates S^{a_1} and S^{a_2} can be trivially commuted through CZ. We are left with the CZ followed by two new local Clifford gates L_1' and L_2' (cf. Fig. 8). Thus we can absorb the CZ in G_i creating $G_{i+1} = G_i \oplus \{v_1 v_2\}$ which is locally preparable on H since, by Proposition 2, $v_1 v_2$ is an edge of H. Moreover we define $C_{i+1} = C_i' L_1' L_2' \in \mathcal{C}_H$. Thus, in *Case 1*, $P(i+1)$ holds.

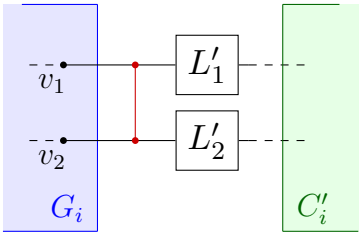

Figure 8: *Case 1*: after commuting L_1 and L_2, creating two new local Clifford gates.

Case 2. Only one of b_1 and b_2 is odd, let us suppose it is b_1 without loss of generality, then L_2 can weakly commute through CZ up to local Cliffords on the other side of the CZ. Likewise, we can always either write L_1 as $\sqrt{X}S$ or $\sqrt{X}^\dagger S$ up to a Pauli gate P that we weakly commute through the CZ, depending on the simplification lemma we want to use. These two rewriting steps result in a circuit in which there are two local Clifford gates L_1' and L_2' after the CZ that can be absorbed back in C_i' to be treated later (cf. Fig. 9).

We only have to write the remaining S and $\sqrt{X}^\dagger$ as byproducts of local complementation operations on vertices of G_i.

First, if v_1 has no neighbor in G_i, then, after applying the S and $\sqrt{X}^\dagger$, qubit v_1 ends up in a product state $|0\rangle$. Consequently, the CZ gate can be removed, and S and $\sqrt{X}^\dagger$ can be absorbed in C_i' to create C_{i+1}.

Otherwise, if v_1 and v_2 are each other's only neighbor in G_i, then we can easily rewrite the subcircuit before C_i' on the qubits v_1, v_2 into a more convenient form, since it is only a

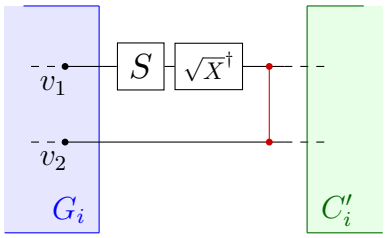

Figure 9: *Case 2*: after commuting a power of S gate and a Pauli gate of the first qubit and L_2 on the second qubit.

two-qubit Clifford circuit that prepares a stabilizer state. Indeed, [VdNDDM04, Theorem 3] provides us, in a constructive manner, an equivalent two-qubit subcircuit that first contains 0 or 1 CZ followed by local Clifford gates. If there is a CZ, we absorb it in G'_i to create G_{i+1}, then we absorb the following single-qubit gates back in C'_i to create C_{i+1}.

Else if the only neighbor of v_1 in G_i is v_2 and v_2 has at least one neighbor in G_i, then we rewrite L_1 as $\sqrt{X}S$ up to a Pauli gate P that weakly commutes through the CZ and we can use Lemma 2 to rewrite the circuit without the CZ gate and absorb the leftover local Clifford gates back into C'_i.

Else v_1 has at least one neighbor u in G_i that is not v_2, then we can use Lemma 3 to rewrite the circuit where the local Clifford gate are after the CZ, then absorb the CZ in G'_i and the leftover local Cliffords into C'_i. So in *Case 2*, $P(i+1)$ holds.

Case 3. Both b_1 and b_2 are odd. Then we can rewrite L_1 and L_2 as $\sqrt{X}^\dagger S$ up to Pauli gates that we commute with the CZ similar to *Case 2*. W.l.o.g., we will treat the qubit of v_1 first.

Now if v_1 has no neighbor in G_i then refer to the analogous sub-case in *Case 2*.

Else if the only neighbor of v_1 in G_i is v_2, then treat the qubit of v_2 first instead. If v_1 and v_2 are each other's only neighbor in G_i then refer to the analogous sub-case in *Case 2*.

Else v_1 has at least one neighbor u in G_i that is not v_2. Let us call W the set of all neighbors of u in G_i that are not v_1. Now we can use the same method of creating pairs of gates that simplify to the identity than before to perform the local complementations $G'_i \leftarrow G_i \star u$, which in particular absorbs the S gate on wire v_1, then $G''_i \leftarrow G'_i \star v_1$, which absorbs the $\sqrt{X}^\dagger$ gate on wire v_1. The rest of the newly created local Clifford gates can be absorbed in C'_i, we call the resulting circuit C''_i (cf. Fig. 10).

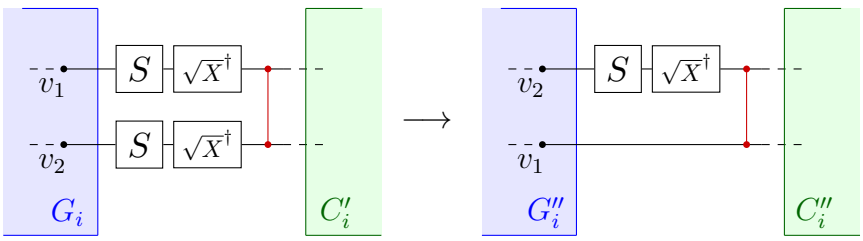

Figure 10: Reducing from *Case 3* to *Case 2*.

We can now use the method described in *Case 2* on v_2 using the graph state G_i'' and the circuit C_i'' to get G_{i+1} and C_{i+1}. In this process, we used only local complementations, thus G_i'' is locally preparable on H. Moreover, as we proved in *Case 2*, only adding gates equivalent to the identity preserved the result $C_i|G_i\rangle = C_i''|G_i''\rangle = C_{i+1}|G_{i+1}\rangle$. So in *Case 3*, $P(i+1)$ holds.

After all the m CZ gates have been absorbed, we have a circuit that prepares a graph state G_m that is pretty much equivalent to G by construction, but to which we applied many local complementation. Thus, now that all the two-qubit gates of C have been processed, all that remains in C_m is a layer of local Clifford gates. As we know that $G_m \sim_{\text{loc}} G$, by [Bou91, VdNDDM04], this layer is equivalent to a sequence of local complementation on vertices $l_1, \ldots, l_s$, thus $G_m \star l_1 \star \cdots \star l_s = G$ is locally preparable on H.

Thus the proposition is true and any graph on n vertices is locally preparable on H. This completes the proof of Theorem 2. □

We remark that in several cases of the proof (e.g., Case 2 subcases 1 and 2, and also thanks to Lemma 2) the CZ count and depth may strictly decrease.

4 Universality and optimality

Informally, Theorem 1 states that any graph state can be prepared on any hardware using only CZ gates and local complementations. Thus this restriction on circuit compilation still gives a universal formalism for graph states.

This method is implemented by a $\mathcal{O}(n^4)$ time constructive algorithm (essentially due to the complexity of Bouchet's algorithm [Bou91] to compute the final sequence of local complementations), that yields an output circuit with a CZ count of at most the CZ count of the input circuit. This allows us to formulate the following theorem.

Theorem 2 (Optimality) *Let $|G\rangle$ be a graph state on at most n qubits and H be a connected graph on n vertices. Among the circuits compilable on H that prepare $|G\rangle$ with optimal CZ count and depth, there exists at least one that consists only of CZ and local complementations.*

Proof. Suppose we have an oracle that yields a circuit C that prepares a given graph state $|G\rangle$ on our target hardware H such that the CZ count of C is optimal. Our construction produces a circuit C' equivalent to C with at most as many CZ as the original circuit. Moreover, these CZ appear in the same order and involve the same sets of qubit. Thus, the optimal CZ count and depth required to prepare any graph state is achieved by circuits using only CZ gates and local complementation, no matter the architecture. □

Interestingly, our result can also be written solely in the language of graph theory, which offers new insight on the expressive power of combining local complementation and edge toggling in graph transformation. Indeed, each CZ gate corresponds to an edge toggling of G compatible with H, and local Cliffords can be grouped into local complementations over G. Thus, our output circuit translates into a sequence of graph operations that can be sequentially applied on the empty graph to produce G. Therefore, we have the following.

Corollary 1 (Graph statement) *Let H be a connected graph on n vertices. Then any graph on n vertices can be created from the empty graph on n vertices with only local complementations and edge toggling in H.*

Remarkably, the proof of this result uses quantum circuit rewriting at its core, and cannot be trivially ported to simple graph theoretic operations only.

5 Discussion and conclusion

Numerical results. We implemented our method and applied our transpilation on random circuits ranging from $n = 6$ to 16 qubits. We first ran the naive qubit routing heuristic using SWAP gates to make the random circuit hardware-compliant. Then, we rewrote this hardware-compliant circuit using our method. Finally, we compared these results to a *loop detection* optimization, that recognizes at every step while the graph state is being prepared if the current graph was previously reached. This allows the deletion of entire subcircuits to return to an equivalent state and effectively shortcuts the graph state preparation, and/or terminates early if the target graph state is reached. We averaged the results on 20 random circuits for each n.

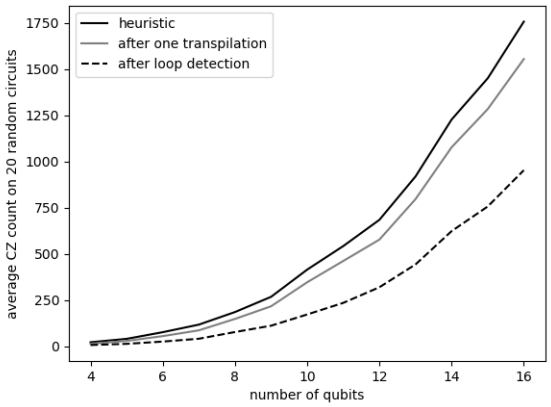

Figure 11: Comparison of the naive heuristic with our method, both with and without the loop detection optimization.

We notice that in the case of (non-optimal) random circuits as the input, our method consistently outputs circuits with a lower CZ count on average. In practice, we find that the loop detection step also allows for CZ count reduction, with the benefit of saving time in the last step of our construction when Bouchet's algorithm is called. We also observed that iterating our method, repeatedly using our transpiled circuit as the input, further reduces the CZ count and depth. We leave the task of implementing a more profound heuristic on the basis of our method for future work.

In this work, we have established a universal framework for graph state preparation on arbitrary quantum hardware constrained by a fixed connectivity graph. By leveraging local complementation's ability to emulate up to a quadratic number of CZ gates ultimately in-

creasing the circuit fidelity, we demonstrated that any graph state can be efficiently prepared while maintaining an upper bound on the number of entangling gates. This result not only provides a new perspective on the synthesis of graph states and stabilizer states but also highlights fundamental graph-theoretic principles underlying quantum state transformations. Whereas other transpilation methods have studied the overhead in CZ/two-qubit gates count to produce a hardware-compliant circuit, we presented a framework to simplify such circuits while preserving the CZ count and depth. We have identified cases where CZ count or depth strictly decrease. It would be interesting to characterize the circuits where it occurs, and precisely quantify this effect. An implementation of our method allow us to experimentally measure the gain in CZ count when given any (not necessarily optimal) input circuit[1].

Practically, our findings provide a systematic method for compiling graph states on near-term quantum devices, ensuring efficient use of limited connectivity. Indeed, graph states can be used as resources for different frameworks such as measurement-based quantum computing (MBQC) [RBB03] or more generally fault-tolerant quantum computing (FTQC) [YR24].

Future work may include finding out whether this result can be further restricted to the use of toggling edges and using pivot operations (triple local complementation on two neighbors in the graph), as well as more graph theory oriented topics, such as using our technique to improve the complexity of Bouchet's algorithm [Bou91] for efficiently computing local complementation sequences in our setting. Additionally, an open question remains whether a purely combinatorial proof of Corollary 1 can be found, further enriching the interplay between graph theory and quantum computation.

References

[AMMR13] M. Amy, D. Maslov, M. Mosca, and M. Roetteler, *A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 32 (2013), p. 818–830. doi: 10.1109/tcad.2013.2244643.

[AMSDS20] J. C. Adcock, S. Morley-Short, A. Dahlberg, and J. W. Silverstone, *Mapping graph state orbits under local complementation*, Quantum, 4 (2020), p. 305. doi: 10.22331/q-2020-08-07-305.

[Bou91] A. Bouchet, *An efficient algorithm to recognize locally equivalent graphs*, Combinatorica, 11 (1991), pp. 315–329. doi: 10.1007/BF01275668.

[BPBB25] S. Brandhofer, I. Polian, S. Barz, and D. Bhatti, *Hardware-efficient preparation of architecture-specific graph states on near-term quantum computers*, Springer Nature / Scientific Reports, 15 (2025). doi: 10.1038/s41598-024-82715-x.

[CCL23] N.-H. Chia, K.-M. Chung, and C.-Y. Lai, *On the need for large quantum depth*, J. ACM, 70 (2023). doi: 10.1145/3570637.

[1] Code available on demand.

[CDLTP11] A. Cabello, L. E. Danielsen, A. J. López-Tarrida, and J. R. Portillo, *Optimal preparation of graph states*, Physical Review A, 83 (2011). doi: 10.1103/physreva.83.042314.

[DKPvdW20] R. Duncan, A. Kissinger, S. Perdrix, and J. van de Wetering, *Graph-theoretic simplification of quantum circuits with the zx-calculus*, Quantum, 4 (2020), p. 279. doi: 10.22331/q-2020-06-04-279.

[DN06] C. M. Dawson and M. A. Nielsen, *The Solovay-Kitaev algorithm*, Quantum Information and Computation, 6 (2006), pp. 81–95. doi: 10.26421/QIC6.1-6.

[DP09] R. Duncan and S. Perdrix, *Graph states and the necessity of euler decomposition*, in Mathematical Theory and Computational Practice, 5th Conference on Computability in Europe (CiE), vol. 5635 of Lecture Notes in Computer Science, Springer, July 2009, pp. 167–177. doi: 10.1007/978-3-642-03073-4_18.

[DW18] A. Dahlberg and S. Wehner, *Transforming graph states using single-qubit operations*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376 (2018). doi: 10.1098/rsta.2017.0325.

[EKC+22] S. Ebadi, A. Keesling, M. Cain, T. T. Wang, H. Levine, et al., *Quantum optimization of maximum independent set using rydberg atom arrays*, Science, 376 (2022), pp. 1209–1215. doi: 10.1126/science.abo6587.

[FL11] S. T. Flammia and Y.-K. Liu, *Direct fidelity estimation from few pauli measurements*, Physical Review Letters, 106 (2011). doi: 10.1103/physrevlett.106.230501.

[FML+17] C. Figgatt, D. Maslov, K. A. Landsman, N. M. Linke, S. Debnath, and C. Monroe, *Complete 3-qubit grover search on a programmable quantum computer*, Nature communications, 8 (2017), p. 1918. doi: 10.1038/s41467-017-01904-7.

[GdBM24] T. Goubault de Brugière and S. Martiel, *Shallower cnot circuits on realistic quantum hardware*, ACM Transactions on Quantum Computing, (2024). doi: 10.1145/3700884.

[GdBMV25] T. Goubault de Brugière, S. Martiel, and C. Vuillot, *A graph-state based synthesis framework for clifford isometries*, Quantum, 9 (2025), p. 1589. doi: 10.22331/q-2025-01-14-1589.

[HBC+22] H.-Y. Huang, M. Broughton, J. Cotler, S. Chen, J. Li, et al., *Quantum advantage in learning from experiments*, Science, 376 (2022), pp. 1182–1186. doi: 10.1126/science.abn7293.

[JL03] R. Jozsa and N. Linden, *On the role of entanglement in quantum-computational speed-up*, Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 459 (2003), pp. 2011–2032. doi: 10.1098/rspa.2002.1097.

[KdG20] A. Kissinger and A. M.-v. de Griend, *Cnot circuit extraction for topologically-constrained quantum memories*, Quant. Inf. Comput., 20 (2020), pp. 581–596. doi: 10.26421/QIC20.7-8-4.

[KKL19] A. Kalev, A. Kyrillidis, and N. M. Linke, *Validating and certifying stabilizer states*, Physical Review A, 99 (2019). doi: 10.1103/physreva.99.042337.

[KM13] V. Kliuchnikov and D. Maslov, *Optimization of clifford circuits*, Physical Review A, 88 (2013). doi: 10.1103/physreva.88.052307.

[KMY25] S. Kumabe, R. Mori, and Y. Yoshimura, *Complexity of graph-state preparation by clifford circuits*, 2025. doi: 10.48550/arXiv.2402.05874.

[Kot68] A. Kotzig, *Eulerian lines in finite 4valent graphs and their transfomations*, Theory of Graphs, (1968), pp. 219–230.

[MMN+16] E. A. Martinez, T. Monz, D. Nigg, P. Schindler, and R. Blatt, *Compiling quantum algorithms for architectures with multi-qubit gates*, New Journal of Physics, 18 (2016). doi: 10.1088/1367-2630/18/6/063029.

[MR18] D. Maslov and M. Roetteler, *Shorter stabilizer circuits via bruhat decomposition and quantum circuit transformations*, IEEE Transactions on Information Theory, 64 (2018), p. 4729–4738. doi: 10.1109/tit.2018.2825602.

[MV06] M. Möttönen[1] and J. J. Vartiainen, *Decompositions of general quantum gates*, Trends in quantum computing research, (2006), p. 149.

[PL23] B. Pokharel and D. A. Lidar, *Demonstration of algorithmic quantum speedup*, Physical Review Letters, 130 (2023). doi: 10.1103/physrevlett.130.210602.

[Pre18] J. Preskill, *Quantum computing in the nisq era and beyond*, Quantum, 2 (2018), p. 79. doi: 10.22331/q-2018-08-06-79.

[RBB03] R. Raussendorf, D. E. Browne, and H. J. Briegel, *Measurement-based quantum computation on cluster states*, Physical Review A, 68 (2003). doi: 10.1103/physreva.68.022312.

[RHK+20] T. Roy, S. Hazra, S. Kundu, M. Chand, M. P. Patankar, and R. Vijay, *Programmable superconducting processor with native three-qubit gates*, Physical Review Applied, 14 (2020). doi: 10.1103/physrevapplied.14.014072.

[Sho97] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Journal on Computing, 26 (1997), p. 1484–1509. doi: 10.1137/s0097539795293172.

[VdNDDM04] M. Van den Nest, J. Dehaene, and B. De Moor, *Graphical description of the action of local clifford transformations on graph states*, Physical Review A, 69 (2004). doi: 10.1103/physreva.69.022316.

[Vid03] G. Vidal, *Efficient classical simulation of slightly entangled quantum computations*, Physical Review Letters, 91 (2003). doi: 10.1103/physrevlett.91.147902.

[WBD+19] K. Wright, K. M. Beck, S. Debnath, J. Amini, Y. Nam, et al., *Benchmarking an 11-qubit quantum computer*, Nature Communications, 10 (2019). doi: 10.1038/s41467-019-13534-2.

[WBLW23] F. Wagner, A. Bärmann, F. Liers, and M. Weissenbäck, *Improving quantum computation by optimized qubit routing*, Journal of Optimization Theory and Applications, 197 (2023), p. 1161–1194. doi: 10.1007/s10957-023-02229-w.

[XLZZ21] Y. Xia, W. Li, Q. Zhuang, and Z. Zhang, *Quantum-enhanced data classification with a variational entangled sensor network*, Physical Review X, 11 (2021). doi: 10.1103/physrevx.11.021047.

[YR24] W. Yang and P. Rall, *Harnessing the power of long-range entanglement for clifford circuit synthesis*, IEEE Transactions on Quantum Engineering, 5 (2024), pp. 1–10. doi: 10.1109/TQE.2024.3402085.

[YZ24] P. Yuan and S. Zhang, *Full characterization of the depth overhead for quantum circuit compilation with arbitrary qubit connectivity constraint*, arXiv preprint arXiv:2402.02403, (2024). doi: 10.48550/arXiv.2402.02403.

[ZCL+22] M.-G. Zhou, X.-Y. Cao, Y.-S. Lu, Y. Wang, Y. Bao, et al., *Experimental quantum advantage with quantum coupon collector*, Research, 2022 (2022). doi: 10.34133/2022/9798679.

[ZHQ+21] C. Zhang, A. B. Hayes, L. Qiu, Y. Jin, Y. Chen, and E. Z. Zhang, *Time-optimal qubit mapping*, in 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Association for Computing Machinery, 2021, p. 360–374. doi: 10.1145/3445814.3446706.

A Proof of Lemma 2

We will prove Lemma 2 using ZX-calculus rewriting rules from [DKPvdW20].

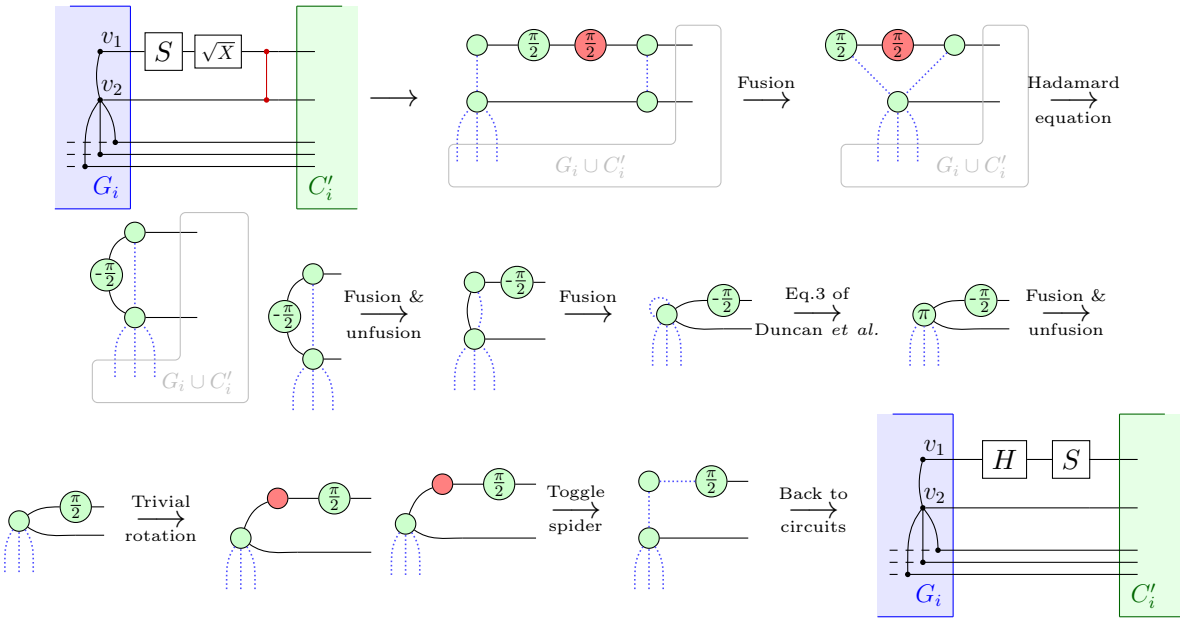

Figure 12: Proof of Lemma 2 based on ZX-calculus.

B Proof of Lemma 3

In this setting, v_1 has at least one neighbor u in G_i that is not v_2. Let us call W the set of all neighbors of u in G_i that are not v_1. W can be the empty set. To reveal a local complementation, we will create a pair of gates $\sqrt{X}^\dagger$ and $\sqrt{X}$ which are equivalent to the identity on u, along with pairs of gates S and $S^\dagger$ that also simplify to the identity on each qubit corresponding to a vertex of W.

This way we will perform the local complementation $G'_i \leftarrow G_i \star u$ by absorbing the $\sqrt{X}^\dagger$ on the qubit of u and the S gates on the qubits of W. The rest of the newly created local Clifford gates can be absorbed in C'_i, we call the resulting circuit C''_i. Notice that if $v_2 \in W$, power of S gates will be created on the second qubit, but those can be commuted with the CZ and left to be treated in a future step.

This first part of the process can be ignored if the power a_1 of the S gate is even: this would be a Pauli gate that would be commuted with the $\sqrt{X}^\dagger$ and the CZ up to a Pauli operator.

Then all that remains between the CZ and G'_i is the $\sqrt{X}^\dagger$ on the qubit of v_1. We will reveal a local complementation by creating pairs of gates S and $S^\dagger$ that simplify to the identity on each qubit corresponding to a neighbor of v_1 in G'_i. Now we can perform the local complementation $G''_i \leftarrow G'_i \star v_1$ by absorbing the $\sqrt{X}^\dagger$ on the qubit of v_1 and the S gates on

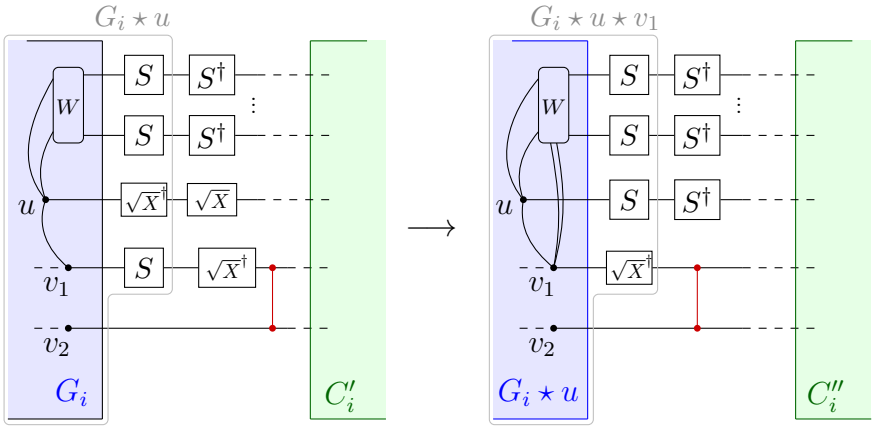

Figure 13: Revealing a local complementation on u by creating pairs of gates equivalent to the identity, then revealing a local complementation on v_1 in the same way.

the qubits of all the neighbors of v_1. The rest of the newly created $S^\dagger$ gates can be absorbed in C''_i potentially commuting through the CZ gate. We can call the resulting circuit C_{i+1}. The only step left to do is to absorb the CZ into G''_i. We can call the resulting graph state $G_{i+1} = G_i \star u \star v_1 \oplus \{v_1 v_2\}$ which is locally preparable on H. Moreover, we only created gates that simplify to the identity thus $C_i|G_i\rangle = C_{i+1}|G_{i+1}\rangle$.