

Measurement-based quantum compiling via gauge invariance

Sebastiano Corli 

*Università degli Studi di Milano, Dipartimento di fisica Aldo Pontremoli, Celoria Street 16, 20133, Milano, Italy,
Politecnico di Milano, Dipartimento di fisica, Leonardo da Vinci Square 32, 20133, Milano, Italy and
Consiglio Nazionale delle Ricerche, Istituto di Fotonica e Nanotecnologie, Leonardo da Vinci Square 32,
20133, Milano, Italy*

Enrico Prati *

*Università degli Studi di Milano, Dipartimento di fisica Aldo Pontremoli, Celoria Street 16, 20133, Milano, Italy
and Consiglio Nazionale delle Ricerche, Istituto di Fotonica e Nanotecnologie,
Leonardo da Vinci Square 32, 20133, Milano, Italy*



(Received 27 January 2025; revised 3 February 2026; accepted 24 March 2026; published 23 April 2026)

The measurement-based architecture is a paradigm of quantum computing, relying on the entanglement of a cluster of qubits and the measurements of a subset of it, conditioning the state of the unmeasured output qubits. While methods to map the gate model circuits into the measurement-based are already available via intermediate steps, we introduce a paradigm for quantum compiling directly converting any quantum circuit to a class of graph states, independently from its size. This method relies on the stabilizer formalism to describe the register of the input qubits. An equivalence class between graph states able to implement the same circuit is defined, giving rise to a gauge freedom when compiling in the MBQC frame. The graph state can be rebuilt from the circuit and the input by employing a set of graphical rules. A system of equations describes the overall process. Compared with measurement calculus, the ancillary qubits are reduced by 50% on the quantum Fourier transform and 75% on the Quantum Approximate Optimization Algorithm, still preserving similar scaling laws for the number of entangling gates.

DOI: [10.1103/lpkt-zgfn](https://doi.org/10.1103/lpkt-zgfn)

I. INTRODUCTION

Measurement-based quantum computing (MBQC) also referred to as one-way quantum computing [1] has become a viable alternative computational paradigm to gate model quantum computing mainly thanks to the development of photonic quantum computers [2–4]. Instead of applying a universal set of unitary operators to a register of initialized qubits, the information processing is carried out after entangling the qubits in a resource state [5,6]. Afterwards, measuring a subset of the resource state, the quantum state of the unmeasured ones (the output qubits) is purposely altered [7–9]. The irreversibility of the measurement justifies the alternative name of one-way quantum computing. In the MBQC frame, a major hindrance consists of entangling all the qubits into a cluster. The main hurdles when entangling a cluster of qubits arise from technological

limitations, e.g., photon loss [8,10]. However, protocols such as entanglement purification, or entanglement witnesses, allow to tackle such withdrawals, achieving a high fidelity on the resource states [11–16]. Next to the generation of the resource state, two-qubit gates are realized by single-qubit measurements during the computation [11,17]. Therefore, once the entangled state has been allocated, it is possible to substitute coherent and long-ranged operations on qubits, envisaging MBQC as a viable architecture in terms of qubit scalability and fault-tolerance [10,18,19]. In MBQC, the entangling gates are applied just before the algorithm executes, thus any error deriving from such process will alter the cluster of entangled qubits, but not necessarily the algorithm itself, which represents a subtle difference to the error correction and mitigation techniques developed for the gate model [10]. Indeed, the latter assumption drops the need to maintain coherent control over the quantum state of qubits, replacing it with the less experimentally challenging goal of initializing a graph state before starting the computation [20,21]. In their proposal, Danos et al. [22,23] provided a method, the MCalculus, to translate a sequence of logic gates into a set of entangling gates, quantum measurements, and Pauli corrections to perform in order to translate the gate sequence

*Contact author: enrico.prati@unimi.it

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

into the MBQC formalism. Such sequence of operations is named the MQBC pattern [23]. Causal flow, gflow, and Pauli flow, for instance, are techniques aiming to optimize the adaptive measurement patterns [24,25]. Other techniques to translate the gate model into the MBQC are exposed in Refs. [26,27].

A cornerstone for clustering graph states into a family of equivalent graphs is provided by LU-equivalence, i.e., a class of graphs which are equivalent up to local unitary transformations [28]. Such transformations are helpful to capture nontrivial quantum properties, related to the entanglement in the system, through a set of simple graphical rules [29]. Nonetheless, even applying local unitary transformations $|\Psi\rangle = \hat{U}_1 \otimes \cdots \otimes \hat{U}_n |\Psi_0\rangle$, the total amount of quantum entanglement remains unaltered [28], an issue which may call for alternative compiling techniques. Local Clifford operations have been employed for efficient compiling techniques, e.g., in Ref. [30], where Clifford operations are carried out in order to minimize the transpiling limited to conversions from cluster states to graph states. Furthermore, specific compiling techniques in the MBQC frame have been addressed for variational quantum eigensolver (VQE) algorithms [31–33], where it has been shown how measurement-based techniques prove to be more efficient than their gate-based counterparts for specific designed Hamiltonians.

Other works adapt Clifford quantum cellular automata (QCA) to MBQC, where QCA form a class of unitaries which are translationally invariant and locality-preserving [34]. A useful protocol to reduce the number of Pauli operations, more specifically the byproduct correction, is the Pauli tracking, which has been already implemented in recent libraries [35–37]. Other frameworks which allow to optimize the size of the cluster states are provided by Ref. [38], whereas other protocols are suited to install fault-tolerant graph states, according to specific error correction codes [39]. Another paramount tool employed for MBQC compiling are the ZX diagrams, a tool which allows the graph states to be described as tensors through a set of rules similar to the Penrose notation [40]. ZX diagrams compiling techniques, however, were limited to measurements in the XY plane. Recently, new MBQC techniques have extended the employment of these diagrams to any direction of measurement [27].

Other methods to convert unitary operations into graph states have been proposed, even without implementing a universal computation, for example, to convert diagonal operators into specific graphs [41]. Recently, Wong et al. [42] formulated an underlying gauge theory over the MBQC, contextualized in the frame of condensed matter and high-energy physics, not related to quantum compiling. Introducing the present work, two short proceedings which sketch the overall method without illustrating proofs and details can be found in Refs. [43,44].

Instead, here we develop an exact method for MBQC compiling, able to map any quantum circuit into a graph state without any transposition from the gate model. The description of the qubit system takes advantage of the notation introduced previously in Ref. [45] and it is described by group theory instead of state vectors or the density matrices. Indeed, the overall state is represented by a set of stabilizers which act on the initialized register $|0\rangle^{\otimes n}$. Such formalism relies on applying a decomposition of the unitary operator in terms of Pauli matrices, whose goal can be accomplished, for instance, via the KAK decomposition [46–51], or rather by any decomposition of a generic unitary in terms of Pauli matrices. The decomposition of the circuit into a set of universal gates could be employed as well, by opportunely handling the rules of Clifford group introduced in the following sections.

Thanks to such decompositions, it is shown that any unitary operator can be converted directly into a graph state, bypassing any circuit conversion from gate-based to measurement-based formalism.

Indeed, we introduce a straightforward construction of the graph relying on a technique which we call *analysis of the residual*. Such technique relies on a set of graphical rules which allow the conversion of a graph state into a linear combination of operators and vice versa. Such an approach allows any unitary, opportunely decomposed, to be mapped directly into a graph state, without transposing the unitary itself through the gate model. Furthermore, as discussed previously, it is known from the literature that multiple graph states, belonging to an equivalence class, are able to apply the same quantum circuit to a specific input state. We extend such an equivalence beyond local operations, giving rise to a gauge freedom, as different graph states, differing not only in their topology but even in their size (i.e., number of nodes), can be chosen without affecting the outcome of the computation. In other words, different graph states may prove to be equivalent even beyond local, single-qubit operations, such as local complementation [52], due to an underlying gauge equivalence which can be deduced by the *analysis of the residual*.

Our method shows that, in order to construct the graph, for mapping any generic unitary $\hat{U}$ representing the quantum circuit into a graph $G[V, E]$, at least one explicit graph representing the solution exists. We refer to this mapping as the fully symmetric gauge, a trivial but illustrative example being provided in the appendices. Such gauge freedom, as well as the overall method, is expressed by a system of algebraic equations. Exploiting the fully symmetric gauge, we cut the number of ancillary qubits, with respect to the MCalculus, by a factor of 1/4 the for Quantum Approximate Optimization Algorithm (QAOA) and 1/2 for the quantum Fourier transform (QFT). As for the total entanglement of the graph state, the results from the two methods are comparable, and they strictly

depend on the instance of the problem (e.g., the topology of the MaxCut graph for the QAOA or the number of layers p to be applied). Nevertheless, the focus of this benchmark is on the fully symmetric gauge, which is universal (it can be applied to any algorithm without methodological variations) and can be considered a fair reference for this method we introduce. Since the MCalculus is a universal method as well, able to transpile any gate-based circuit into a graph state, a worthy benchmark can be established between these two mappings. However, more gauges can be inquired developing numerical approaches or theoretical studies, even algorithm-based, and future developments could compare more recent, local-equivalent approaches with more sophisticated and

algorithm-based gauges. The main purpose of this work remains introducing a new formalism able to rearrange MBQC compiling through a gauge freedom. Eventually, the formulation of MBQC compiling via a system of equations allows to invert the already acknowledged methods for mapping unitary operators into a graph state.

II. RESULTS

The MBQC paradigm consists of entangling a cluster of qubits and progressively measuring a subset of resource qubits, whose entanglement with the unmeasured qubits conditions the output state of the latter ones [Fig. 1(a)]. The rationale behind the direct compiling method, which

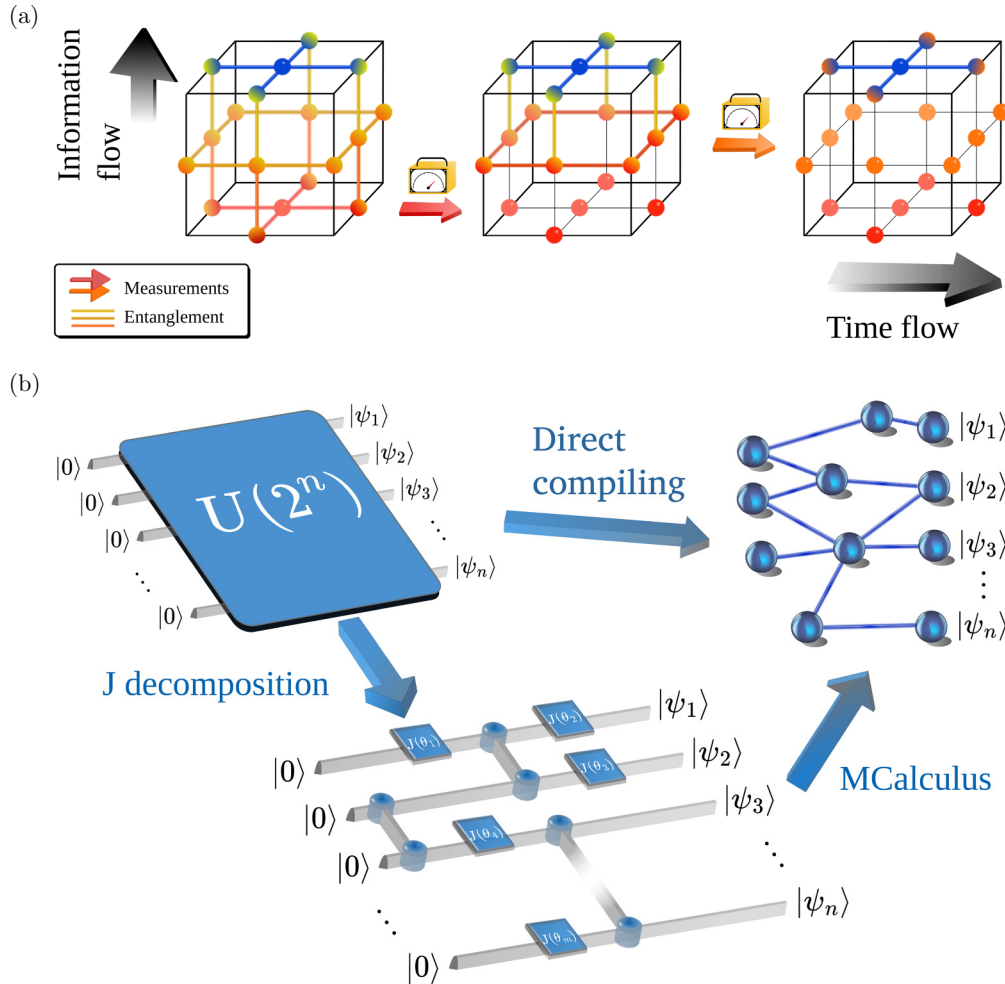


FIG. 1. (a) In the MBQC paradigm, the graph state (here represented by two nodes linked by the violet line, representing their entanglement) is split into the resource state, i.e., the qubits which have to be measured to let the information flow (in a blue and red superposition) and the output states (red and blue superposition). Due to the entanglement, the output qubits are affected by the measurements over the resource qubits. (b) In quantum compiling, a generic unitary operator needs to be decomposed into elementary operations. In the measurement-based architecture, the unitary operation is prompted by entangling the resource states with the output qubits, thereafter measuring the previous ones. Currently, the process consists of two steps, namely J decomposition of the unitary operator and then the construction of the graph state by using MCalculus. The aim of this work is to build a compiling algorithm to cast directly the unitary matrix from $\hat{U}(2^n)$ into the graph state.

prepares the starting graph state from any given arbitrary unitary operator, representing a quantum circuit, discussed in this section is represented in Fig. 1(b). The demonstration of the validity of this compiling method is demonstrated by construction through three major steps. The first step consists of the description of the input state, by merging the circuit and the state of the register by a single operator, later called circuit endomorphism. Although it would be possible to rely on the commonly adopted wording, it is required to develop the following Heisenberg picture of the stabilizer states in order to make manifest the gauge freedom, the key feature of this work. The second step consists of preparing the equivalence class which leaves the output state unchanged after having consistently altered both the input circuit operator and the input state (expressed, in turn, as an operator). Finally, as a third step, such gauge invariance provides the MBQC embedding by graphical rules, defining an equivalence class of graphs, from which the MBQC algorithm starts to be computed. The overall picture of the MBQC compiling is expressed by a system of core equations.

A. Preparation of the input state

We start by describing the first step, consisting of merging the given circuit with the register of input qubits. First, the input circuit $\hat{U}$ is expressed by its decomposition into rotations induced by elements of the Pauli algebra. Next, the input state is expressed by operators $\hat{G}_i$ called Pauli endomorphisms (PEs), i referring to each qubit, instead of the usual vector state. These operators map the $|0\rangle^{\otimes n}$ state into any state $|C\rangle$ available by employing only Clifford operators. Therefore, the $\hat{G}$ operators act as endomorphism inside the Hilbert space $\mathcal{H}^{\otimes n}$.

Eventually, the composition of the circuit $\hat{U}$ with the PEs $\prod_i \hat{G}_i$ gives rise to the circuit endomorphism η , a tensor representation of the original circuit and input register by a single operator $\hat{U} \prod_i \hat{G}_i$.

1. Pauli decomposition.

The input circuit $\hat{U}$ of n qubits is first represented by rotations induced by elements from the Pauli group $\hat{\sigma} \in \mathcal{P}$. Such representation can be borrowed, e.g., from the KAK decomposition:

$$\begin{aligned} \hat{U} &= \hat{U}_l \hat{U}_{l-1} \cdots \hat{U}_1 \\ &= [\cos(\theta_l) + i \sin(\theta_l) \hat{\sigma}_l] \cdots [\cos(\theta_1) + i \sin(\theta_1) \hat{\sigma}_1]. \end{aligned} \quad (1)$$

Each of the $\hat{U}_i$ refers to a specific operation described in a quantum circuit, which may involve a number of qubits ranging from 1 to n .

2. Expression of the input state as PEs.

A suitable formalism to describe the input state is introduced. Given a register $|\Psi\rangle$ of n qubits, $|\Psi\rangle \in [\mathbb{C}^2]^{\otimes n}$, the PE $\hat{G}_i$ of the i th qubit is defined as a linear combination of the identity $\hat{I}_{N \times N}$ ($N = 2^n$) plus a Pauli operator $\hat{K}_i = \hat{X}_i \otimes_{j=1}^n \hat{\sigma}_j$ which we call *stabilizer*, where $\hat{\sigma}_j \in \{\hat{I}_{2 \times 2}, \hat{Z}_j\}$, the latter being a subgroup of the Pauli group $\mathcal{P}_1$:

$$\hat{G}_i(a, b) = a \hat{I}_{N \times N} + b \hat{K}_i. \quad (2)$$

From now on, to ease notation, $\hat{I}_{N \times N}$ will be omitted. In a dedicated lemma from Appendix A, we prove that the PEs $\hat{G}$ are not unitary, even though returning $\langle 0 | \hat{G}^\dagger \hat{G} | 0 \rangle = 1$. Any quantum state encoding a single qubit can be represented by

$$|\psi\rangle = a |0\rangle + b |1\rangle = [a + b \hat{X}] |0\rangle = \hat{G}(a, b) |0\rangle. \quad (3)$$

A register of n nonentangled qubits can be generalized as

$$|\Psi\rangle = \prod_{i=1}^n [a_i + b_i \hat{X}_i] |0\rangle^{\otimes n} = \prod_{i=1}^n \hat{G}_i(a_i, b_i) |0\rangle^{\otimes n}. \quad (4)$$

The coefficients in front of the $\hat{K}_i$ operator (the b_i) are from now on referred to as the *interactive coefficients* of the PE $\hat{G}_i$. The coefficients in front of the $\hat{I}_{N \times N}$ identity (the a_i) are referred to as the *idle coefficients* of the $\hat{G}_i$ PE. The entanglement between two PEs $\hat{G}_i, \hat{G}_j$ can be induced by the $\hat{C}Z_{ij}$ phase operator, morphing their stabilizer $\hat{X}_i \xrightarrow{\hat{C}Z_{ij}} \hat{X}_i \hat{Z}_j = \hat{K}_i$. A qubit i entangled with a set $\mathcal{N}(i)$ of qubits displays a stabilizer such as

$$\hat{K}_i = \hat{X}_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j, \quad (5)$$

with $\mathcal{N}(i)$ representing the sequence of all the $\hat{C}Z_{ij}$ Clifford operations. In Appendix B, we show, as a quick reminder, that the $\hat{K}_i, \hat{K}_j$ stabilizers still commute each other with, even after applying the $\hat{C}Z_{ij}$ phase gates. In Appendix C, we summarize the rules for the Clifford algebra, which $\hat{C}Z_{ij}$ belongs to. Therefore, when dealing with MBQC, the PEs allow us to implement the overall system in a much more compact representation than by installing a state vector.

3. Construction of the circuit endomorphism tensor.

We define the *circuit endomorphism* $\hat{\eta}$ as the composition between the product of the PEs $\prod_i \hat{G}_i$ and the unitary

implementing the quantum circuit:

$$\hat{\eta} = \hat{U} \prod_{i=1}^n \hat{G}_i. \quad (6)$$

The circuit endomorphism, as a $2^n \times 2^n$ matrix, can be expressed in a tensor formalism by the canonical basis $\hat{e}_i$ of the 2×2 matrices:

$$\hat{\eta} = \eta^{i_1 \dots i_n} \hat{e}_{i_1} \otimes \hat{e}_{i_2} \otimes \dots \otimes \hat{e}_{i_n},$$

$$\hat{e}_i \in \left\{ \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \right\}, \quad (7)$$

where we followed the Einstein notation on the indices.

B. Definition of the equivalence class

Once the input register and the quantum circuit are unified into the circuit endomorphism tensor, we now turn to the definition of equivalence class of the circuit endomorphism operators. In order to natively adopt operatorial expressions ready for the next step of the compiling process, we aim to describe the quantum circuit and the register as a linear combination of elements from the Pauli algebra. Therefore, in order to define the equivalence class, we start by expressing the circuit endomorphism tensor in the Pauli basis. Next, the equivalence class is established between those combinations of quantum circuits $\hat{U}$ and the register $\prod_i \hat{G}_i$ which act identically over $|0\rangle^{\otimes n}$, i.e., which are characterized by a circuit endomorphism with identical first column. Eventually, we introduce a map between elements of the same equivalence class, which constitutes the gauge transformations exploited in the final step of the compiling method.

1. The circuit endomorphism in the Pauli basis.

Due to Eqs. (1) and (4), the circuit endomorphism $\hat{\eta}$ can also be described as a linear combination with generators from the Pauli group $\sigma \in \{\hat{I}, \hat{X}, \hat{Z}, \hat{Z}\hat{X}\}$. Such linear combination in the Pauli basis is referred to as the *transfer tensor* $\hat{T}$:

$$\hat{T} = T^{i_1 \dots i_n} \hat{\sigma}_{i_1} \otimes \hat{\sigma}_{i_2} \otimes \dots \otimes \hat{\sigma}_{i_n}. \quad (8)$$

The transfer tensor and the circuit endomorphism describe both the same tensor object but for a change of basis, implemented by an isomorphism M :

$$\eta_{i_1 \dots i_n} = M_{i_1 j_1} \dots M_{i_n j_n} T_{j_1 \dots j_n}. \quad (9)$$

2. Criterion to establish the equivalence class.

By construction, the action of the circuit endomorphism is expressed over the $|0\rangle^{\otimes n}$ state:

$$\hat{\eta} |0\rangle^{\otimes n} = \vec{\eta}, \quad (10)$$

where $\vec{\eta}$ is the first column of the $2^n \times 2^n$ matrix implemented by $\hat{\eta}$. From now on, $\vec{\eta}$ is referred to as *vector circuit endomorphism* (which we shorten to *vector circuit*).

3. Equivalence class between circuits.

As the output of the computation is determined by the sole $\vec{\eta}$ vector circuit, as stated by Eq. (10), a class of equivalence between different circuit endomorphisms η, η' is established, once the criterion of equivalence is set by the condition $\vec{\eta} = \vec{\eta}'$:

$$\hat{\eta} \sim \hat{\eta}' \Leftrightarrow \vec{\eta} = \vec{\eta}'. \quad (11)$$

Transforming the circuit endomorphism, and therefore the circuit itself, without spoiling the output of the computation gives rise to a gauge transformation, i.e., a map between different elements of the aforementioned equivalence class. As we are interested in quantum computation, expressed by Pauli operators, we implement such gauge transformation on the transfer tensor:

$$T'_{i_1 \dots i_n} = S_{i_1 \dots i_n}^{j_1 \dots j_n} T_{j_1 \dots j_n} \quad (12)$$

or, by shortening notation,

$$T'_i = S_{ij} T_j, \quad M_{i_1 j_1} \dots M_{i_n j_n} = M_{ij}. \quad (13)$$

We refer to the tensor S as the *gauge tensor* (Fig. 2). Collecting all the information from Eqs. (9), (10), and (12), the following system holds:

$$T'_i = S_{ij} T_j = S_{ij} M_{jk} \eta_k, \quad (14a)$$

$$\vec{\eta} = \vec{\eta}', \quad (14b)$$

$$\eta'_i = M_{ij}^{-1} T'_j = M_{ij}^{-1} S_{jk} \eta_k. \quad (14c)$$

C. MBQC graphic rules and core equations

After the equivalence class is established, we explicitly refer to the MBQC framework by imposing that the linear combination of Pauli operators (expressed by the transfer tensor) is the result of a measurement process over a graph state. Therefore, we build a special transfer tensor, suitable for MBQC, implementing the action of the measurement process by defining a to-be-called residual operator. Next, we apply topological rules expressed by two major theorems. Symmetry constraints are imposed to select those transfer tensors suitable for MBQC compiling and thus generating a graph state, associated to the original input circuit. This MBQC compiling method is embodied by a system of core equations.

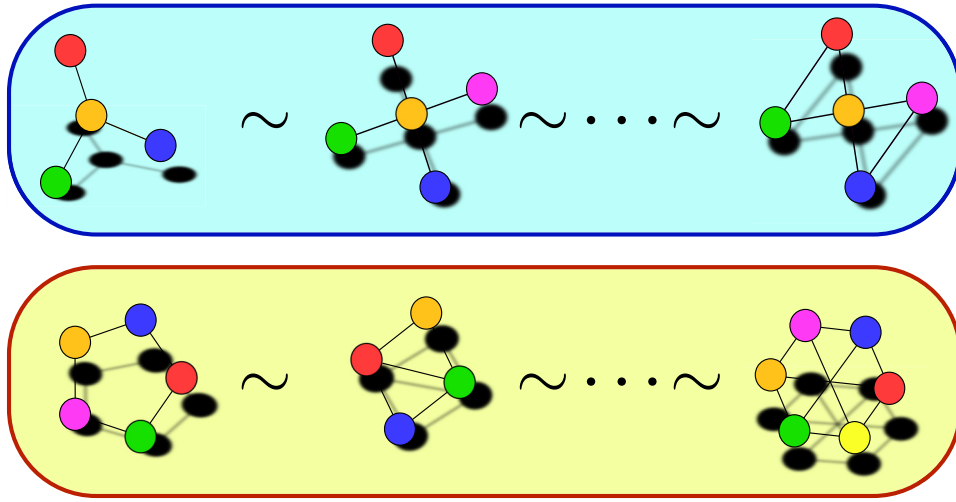


FIG. 2. Graphical representation of two equivalence classes. All of the elements inside a class share a common $\vec{\eta}$ vector circuit. The maps between graphs belonging to the same class are implemented by the S gauge tensor. Note that the color code for the nodes in the graph does not encode any physical meaning.

1. Measurements in the PE formalism.

A measurement on the i th qubit is implemented by a projector operator $\hat{P}_\gamma = |\gamma_i\rangle \langle \gamma_i|$, where $|\gamma_i\rangle$ is the eigenstate of an observable $\hat{\sigma}$. After the measurement of the i th qubit is performed, the measured qubit lies in the $|\gamma\rangle_i$ state. To simplify the notation, we write down the measurements as $\langle \gamma_i|$, instead of $|\gamma_i\rangle \langle \gamma_i|$. A measurement over a generic state $\langle \gamma_i|$ can be represented as

$$\langle \gamma_i| = \alpha_i^* \langle 0_i| + \beta_i^* \langle 1_i| = \langle 0_i| [\alpha_i^* + \beta_i^* \hat{X}_i] = \langle 0_i| \Gamma(\alpha_i, \beta_i), \quad (15)$$

with Γ being the PE for such measurement, and $\alpha, \beta \in \mathbb{C}$, $|\alpha|^2 + |\beta|^2 = 1$. Here Γ involves no $\hat{Z}$ Pauli operator as the measurements act as projections over a single-qubit Hilbert space, with no entanglement involved. In the literature, the measurements with respect to the Pauli observables can be performed over three planes, XY , XZ , and YZ [7,53,54], each of them being a linear combination of two Pauli matrices, whose coefficients are given as $\cos(\theta)$, $\sin(\theta)$. In Appendix D, we explain how to represent such projections in the PE representation.

2. The residual operator.

The action of the measurements over an input state $|\Psi\rangle$ can be therefore described by an operator $\hat{R}$, which we call the *residual* of the measurements, or simply the residual. Such operator is nothing but the PE of a qubit transformed after the measurement itself, i.e., after projecting the single-qubit state described by the PE over a subspace as in Eq. (15). Given a state $|\Psi\rangle$ of $n + m$ qubits, the residual $\hat{R}$ is the operator acting over the new n -qubits state $|\Psi'\rangle$, after a subset of m qubits has been measured,

according to

$$\begin{aligned} & m^\otimes \langle \gamma_m| \prod_{j=1}^m [a_j + b_j \hat{K}_j] \prod_{l=m+1}^{n+m} [a_l + b_l \hat{K}_l] |0\rangle^{\otimes(n+m)} \\ &= \hat{R} \prod_{l=m+1}^{n+m} [a_l + b_l \hat{K}'_l] |0\rangle^{\otimes n} \end{aligned} \quad (16a)$$

$$\hat{R} = m^\otimes \langle \gamma_m| \prod_{j=1}^m [a_j + b_j \hat{K}_j] |0\rangle^{\otimes m} \quad (16b)$$

where the $\langle \gamma_i| = \langle 0| [\alpha_i^* + \beta_i^* \hat{X}_i]$ are the measurements over the $|\Psi\rangle$ state, the latter being allocated by the $n + m$ PEs. After the measurements, the new $|\Psi'\rangle$ state is defined by n new $\hat{K}'_l$ PEs, determined by the previous measurements, whereas the previous m PEs $\hat{K}_j$ are transformed into the $\hat{R}$ residual. In physical terms, the transformation $\hat{K}_l \rightarrow \hat{K}'_l$ means that the entanglement between the unmeasured qubits and the measured ones has been severed. In fact, the n stabilizers $\hat{K}_l$ of the unmeasured qubits transform as follows:

$$\hat{K}_l = \hat{X}_l \bigotimes_{j \in \mathcal{N}(l)} \hat{Z}_j \rightarrow \hat{K}'_l = \hat{X}_l \bigotimes_{j \in \mathcal{N}(l) \setminus \mathcal{M}} \hat{Z}_j, \quad (17)$$

where $\mathcal{N}(l)$ denotes the set of the qubits entangled with the l th one before the measurements and $\mathcal{M}$ denotes the set of measured qubits. In order to tackle the expression of the residual in Eq. (16b), all the $|0\rangle^{\otimes m}$ kets must be contracted to compute the expectation value. As $\hat{Z}|0\rangle = |0\rangle$, the action of such contraction over the PEs can be summed up as follows: suppose p is the qubit to be measured, any

$\hat{G}_l$ PE thus transforms into a $\hat{G}'_l$ as

$$\begin{aligned}\hat{G}_l |0_p\rangle &= \left[a_l + b_l \hat{X}_l \bigotimes_{j \in \mathcal{N}(l)} \hat{Z}_j \right] |0_p\rangle \\ &= |0_p\rangle \left[a_l + b_l \hat{X}_l \bigotimes_{j \in \mathcal{N}(l) \setminus \{p\}} \hat{Z}_j \right] = |0_p\rangle \hat{G}'_l, \quad (18)\end{aligned}$$

proving the validness of Eq. (17). Furthermore, the residual $\hat{R}$, over a register of n qubits, is expressed as a linear combination of $\bigotimes_{j=1}^n \hat{k}_j$ operators, with $\hat{k}_j \in \{\hat{I}_{2 \times 2}, \hat{Z}\}$. The residual is yielded by the following expectation value over the i th qubit:

$$\begin{aligned}\hat{R} &= \langle 0_i | [\alpha_i^* + \beta_i^* \hat{X}_i] \left[a_i + b_i \hat{X}_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j \right] | 0_i \rangle \\ &= \alpha_i^* a_i + \beta_i^* b_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j = A_i + B_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j. \quad (19)\end{aligned}$$

To prove such statement, refer to Appendix E. Consequently, in the MBQC frame the action of a unitary operator can be expressed into a linear combination of $A_i + B_i \bigotimes_{j=1}^n \hat{k}_j$ operators. For the sake of clarity, we may now introduce two definitions. Any coefficient $A_i \neq 0$ from the residual $\hat{R}$, as in Eq. (19), is labeled as an *idle coefficient*. Instead, any coefficient B_i from the residual $\hat{R}$, as in Eq. (19) and $\mathcal{N}(i) \neq \emptyset$, is called an *interactive coefficient*.

It is now possible to enunciate the two theorems mentioned previously. The first theorem states that, after a measurement, the interactive coefficients from the PEs are multiplied in the interactive coefficient of the residual, and separately the idle coefficients from the PEs are multiplied in the idle coefficient of the residual, with no cross-terms. In the second theorem, we state how two measured qubits, previously entangled, output a minus sign multiplying their interactive coefficients in the residual, a property which is a key element of the compiling algorithm. Indeed, from the second theorem a corollary follows which is exploited, along with the corresponding theorem, to build the compiling algorithm for MBQC.

3. Coefficients theorem.

After a $\langle \gamma_i |$ measurement, $\langle \gamma_i |$ in the form of Eq. (15) and $i \in \{0, \dots, n\}$, the interaction coefficients b_i, β_i^* and the idle coefficients a_i, α_i^* , where α_i, β_i are from the PE $\hat{\Gamma}$ of the measurement and a_i, b_i from the PE $\hat{G}$ of the measured qubit, are separately multiplied into two distinguished idle and interaction coefficients A_i and B_i of the residual. The proof is provided by Appendix F.

4. Sign theorem.

Given a register of N qubits, with qubits i and j entangled each other and generated respectively by $\hat{G}_i(a_i, b_i)$

and $\hat{G}_j(a_j, b_j)$, after their measurement the residual $\hat{R}$ displays a term with their interacting coefficients B_i, B_j , a minus sign in front of it and a set of operators $\hat{Z}_k \in \mathcal{N}(i, j) = \mathcal{N}(i) \cup \mathcal{N}(j) \setminus \mathcal{N}(i) \cap \mathcal{N}(j) \setminus \mathcal{M}$, where such set denotes the unmeasured qubits previously entangled with either the i th or the j th site:

$$-B_i B_j \bigotimes_{k \in \mathcal{N}(i, j)} \hat{Z}_k, \quad (20)$$

and $\mathcal{M}$ still denotes the set of the measured qubits. The proof lies in Appendix G.

5. Operator corollary.

When a single interactive coefficient i holds in a term from the residual, the operators in front of it are set as $\bigotimes_{k \in \mathcal{N}(i) \setminus \mathcal{M}} \hat{Z}_k$:

$$A_1 \cdots A_{i-1} B_i A_{i+1} \cdots A_n \bigotimes_{k \in \mathcal{N}(i) \setminus \mathcal{M}} \hat{Z}_k. \quad (21)$$

The proof of the corollary is provided in Appendix H. The two theorems and their corollary provide a set of rules on how to map a graph into a sequence of operators, a graphic approach which could ease the computation. Figure 3 portrays a practical example where such rules are applied. A practical application of the two theorems along with the corollary is provided in Appendix I. A practical example of an application of the two theorems is provided in Fig. 4 of Appendix I.

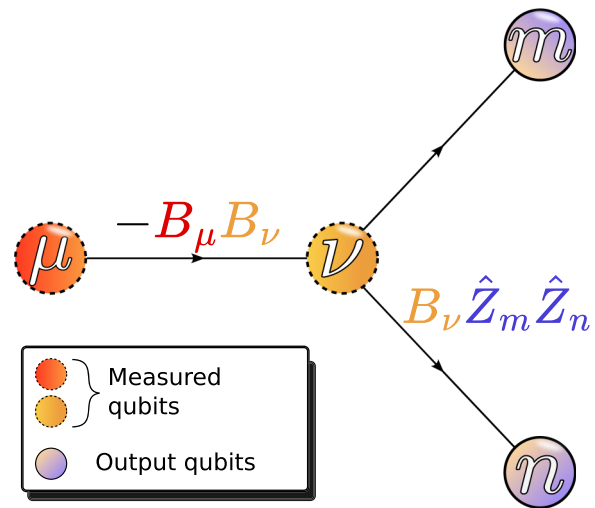


FIG. 3. Graphical rules which link the residual operator to the graph states. When two resource qubits are neighbors (Greek letters, dashed borders), a minus sign occurs in front of their interactive coefficients, whereas a $\hat{Z}$ operator is used when a measured qubit neighbors with the output m, n ones. The computational flow follows the arrows on the edges of the graph.

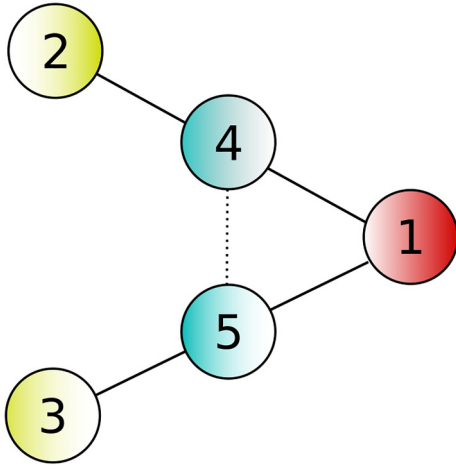


FIG. 4. Two qubits 4 and 5 (in light blue) are measured. The yellow qubits are entangled with 4 or 5, whereas the red qubit is entangled with both of them. After the measurements, the PE of 1 switches from $\hat{X}_1\hat{Z}_4\hat{Z}_5 \rightarrow \hat{X}_1$ and so on for 2 and 3. The interactive coefficients of 4 and 5 in the residual, therefore, yield the $(-)^s B_4\hat{Z}_1\hat{Z}_2 B_5\hat{Z}_1\hat{Z}_3$ term. As both the qubits are entangled with the 1th one, $1 \in \mathcal{N}(4) \cap \mathcal{N}(5)$, and therefore no $\hat{Z}_1$ operator appears. If the dashed line were a solid one, i.e., the two qubits were entangled, it holds that $s = 1$, otherwise $s = 0$ and no line appears between 4 and 5. The qubits 2 and 3 (yellow) belong to $N(4,5) = N(4) \cup N(5) \setminus \mathcal{N}(4) \cap \mathcal{N}(5)$, whereas the 1st one (red) belong to $\mathcal{N}(4) \cap \mathcal{N}(5)$, and thus it is erased from the B_4B_5 term.

6. Considerations on the dimensionality.

We now ponder where the curse of dimensionality may affect this formalism. Writing down the operators in the Heisenberg picture, this action does not entail overheads in the memory storage, since any operator can be settled in the Heisenberg representation as $a_i + b_i\hat{\sigma}$ (the $\hat{\sigma}$ operator belonging to the Pauli group $\mathcal{P}$). Therefore, for any gate displaced in a quantum circuit, its Heisenberg representation follows with ease. The curse of dimensionality arises when computing the residual, as can be noted from Appendix I. Any measurement doubles the length of the residual, whose size increases by 2^m , m being the number of measurements. Nevertheless, for the analysis of the residual, only the terms involving the B_i and the B_iB_j coefficients are needed, i.e., a total number of $n + n(n+1)/2 = O(n^2)$ elements. Since the B_i terms are required to study the connections between the resource qubits to be measured and the output ones, whereas the B_iB_j are used to inquire the connections between the resource qubits themselves, no further terms are required when expanding the residual operator.

7. Symmetries on the transfer tensor.

When implementing a single-qubit operation, the graph state holds a single output qubit, whereas the other qubits

in the cluster remain to be measured. Although the output qubit can be represented via its PE, the measured qubits can be expressed in the residual. For a single-qubit system, the PE is given by $\hat{G}_1(a,b) = [a + b\hat{X}_1]$, where the index 1 referring to the single qubit can be therefore omitted. In the PE there are no $\hat{Z}_i$ terms, as no further entangled qubits i are involved. Therefore, following from Eq. (19), in the residual $\hat{R}$ only a $\hat{Z}$ operator is present as $\mathcal{N}_i$ consists of the sole output qubit. The PE and the residual are thus explicitly expressed as

$$\hat{R}|\Psi\rangle = [A + B\hat{Z}][a + b\hat{X}]|0\rangle, \quad (22)$$

where A and B are $\mathbb{C}$ -numbers which collect all the coefficients from the residual, whereas a and b are the coefficients which form the PE $\hat{G}$ over $|0\rangle$, i.e., the output qubit. From here we introduce the circuit endomorphism $\hat{\eta}$, explicitly expressed as

$$\hat{\eta} = \begin{bmatrix} Aa + Ba & Ab + Bb \\ Ab - Bb & Aa - Ba \end{bmatrix}. \quad (23)$$

The transformed state $|\psi'\rangle$ is described by the vector circuit $\vec{\eta}$, consisting of the first column of the circuit endomorphism:

$$\hat{\eta}|0\rangle = \vec{\eta}, \quad \vec{\eta} = \begin{bmatrix} Aa + Ba \\ Ab - Bb \end{bmatrix}. \quad (24)$$

The corresponding object in the Pauli basis, i.e., the transfer tensor, reads as

$$\hat{T} = Aa\hat{I} + Ab\hat{X} + Ba\hat{Z} + Bb\hat{Z}\hat{X} = (Aa, Ab, Ba, Bb). \quad (25)$$

Even though all of the residuals $\hat{R}$ from a graph state correspond to a unitary operation, or at least to a linear combination of $\hat{Z}_i$ operators, the reverse condition does not necessary hold, i.e., a generic unitary $\hat{U}$ cannot be straightforwardly decomposed as a linear combination of $\hat{Z}_i$ lacking a proper rearrangement. Such a hurdle prompts us to study under what conditions the MBQC computation can be performed, and eventually leads us to focus on the symmetries on the transfer tensor. Such symmetry can be expressed as

$$T_0T_3 = T_1T_2 \Rightarrow AaBb = AbBa. \quad (26)$$

The swap between the a, b coefficients can be implemented, in terms of indices, by a finite and discrete group $(i, \circ)$, whose operation $\circ$ implements the transformations in Table I. Thanks to the action of such group, Eq. (26) can be reformulated as follows:

$$T_iT_j = T_{i\circ j}T_{j\circ i}. \quad (27)$$

It is quite straightforward to show that, except for $0 \circ 3$, $3 \circ 0$, $1 \circ 2$, and $2 \circ 1$, all the other labels yield a simple

TABLE I. Group operations $\circ$ for the symmetries on transfer tensor T . How the labels i, j of the transfer tensor $\hat{T}$ transforms under exchange of the a, b idle and interactive coefficients, the second being linked to the $\hat{X}$ Pauli operator. Such operation is expressed explicitly in Eq. (26). The $i \circ j = k$ operation is summarized in the table: the first column collects the i terms, the first row the j terms, whereas the crossed cells show the final product k . All the operations consist of a swap or an identity operation, with respect to the i th and the j th elements, except for those on the antidiagonal (in bold).

$\circ$	0	1	2	3
0	0	1	0	1
1	0	1	0	1
2	2	3	2	3
3	2	3	2	3

identity, i.e., $T_i T_j = T_j T_i$. Such achievement is summarized in Table I: while almost every couple of elements is closed with respect to the $\circ$ operation, e.g., $i, j \rightarrow j, i$ or the identity $i, j \rightarrow i, j$, the coefficients on the antidiagonal (in red) overturn such rule. When dealing with an n -qubit register, Eq. (27) must be rearranged properly for the n -qubit transformations. The residual of the measured qubits, providing a $\hat{Z}_i$ term for all the output qubits along with the $\hat{Z}_i \hat{Z}_j$ combinations:

$$\hat{\eta}_n = \sum_{m=0}^n \sum_{\substack{\vec{i} \subset [1, \dots, n] \\ |\vec{i}|=m}} \left(C_{\vec{i}} \otimes \hat{Z}_{\vec{i}} \right) \prod_{j=1}^n \hat{G}(a_j, b_j), \quad (28)$$

i.e., the sum runs over all the possible combinations of $\hat{Z}_l$ operators. For the case $n = 2$ in Appendix J, $\vec{i}$ varies to be $\emptyset$, $[1]$, $[2]$, and $[1, 2]$, raising the C_0 , C_1 , C_2 , and C_{12} coefficients. The PEs are supposed to install nonentangled qubits, therefore $\hat{G}(a_i, b_i) = a_i + b_i \hat{X}_i$. The C_i coefficients are $\mathbb{C}$ -numbers, which collect all the other coefficients, as follows from the sign theorem and the operator corollary. The formalism from Table I needs to be extended for the general case n . Leaving the proof in the methods, the composition operation translates into the following expression:

$$T_i T_j = T_{i \circ j} T_{j \circ i} = T_{(i_1, i_2, \dots, i_n) \circ (j_1, j_2, \dots, j_n)} T_{(j_1, j_2, \dots, j_n) \circ (i_1, i_2, \dots, i_n)}, \quad (29)$$

the expression of which involves all the possible compositions between $(i_1, i_2, \dots, i_n) \circ (j_1, j_2, \dots, j_n)$ and vice versa between $(j_1, j_2, \dots, j_n) \circ (i_1, i_2, \dots, i_n)$. Section IV B sets a rigorous proof, whereas a didactic example is provided for the $n = 2$ case in Appendix J.

8. Core equations of MBQC compiling

A system of equations is eventually formulated, by adding Eq. (29) into the already encoded system of

Eqs. (14):

$$T'_i = S_{ij} T_j = S_{ij} M_{jk} \eta_k, \quad (30a)$$

$$\eta_{i0} = \eta'_{i0}, \quad (30b)$$

$$\eta'_i = M_{ij}^{-1} T'_j = M_{ij}^{-1} S_{jk} T_k \quad (30c)$$

$$T_i T_j = T_{i \circ j} T_{j \circ i}, \quad (30d)$$

where $M_{ij} = M_{i_1 j_1} \cdots M_{i_n j_n}$ is the generalized tensor for the change of basis. The importance of these equations is summarized in the pseudocode 1. First, the unitary operator must be decomposed into operators which suit a representation of the form $a_i + b_i \hat{K}_i$. Afterwards, the corresponding representation of the transfer tensor T can be built. Once the representation of T is provided, it is possible to plug it into the above system of equations to query the most suitable gauge S_{ij} , which returns a new T' tensor from Eq. (30a). However, the new transfer tensor corresponds to a graph state if and only if Eqs. (30b) and (30d) are satisfied. While Eq. (30b) checks whether the logic operation we implement is left invariant (i.e., we are operating a gauge transformation), Eq. (30d) verifies if the logic operation physically correspond to an implementable graph state. The mapping from T' to η'_{i0} (as well as from T to η_{i0}) can be accomplished by Eq. (30c). Once a new suitable transfer tensor T' is provided, the topology of the graph state directly descends from the graphic rules established by the theorems in Sec. II C. Note that if we set the same register of qubits in a different state, the compiling process is not altered, since the value of the idle and interactive coefficients a_i, b_i does not carry any topological information.

III. DISCUSSION

A. From graphs to the residual and vice versa

It is therefore possible to introduce a set of techniques to rebuild the graph state from the residual of the unitary operator. First, a decomposition of the unitary matrix $\hat{U}$ has to be given in terms of Pauli matrices, such as KAK decomposition or any composition of universal gates. Once such Pauli decomposition is provided, it is possible to transcribe the transfer tensor. However, the resulting outcome may not fit into the relation expressed by Eq. (30d), which requires a gauge transformation to be applied, in order to satisfy both Eqs. (30d) and (30b). Techniques to be applied over a specific gauge transformation, which we call the fully symmetric gauge, are introduced in Appendix K. A practical example to exploit the fully symmetric gauge is illustrated in Appendix L, whereas an alternative technique, employing solely local Clifford transformations, is given in Appendix M. A class of equivalence between graphs corresponding to the same unitary via different gauge transformations is stated in

ALGORITHM 1. Gauge Compiling.

-
-
- 1: Decompose $\hat{U}$ unitary operator $2^n \times 2^n$ into Pauli basis $\{\hat{I}, \hat{X}_i, \hat{Z}_i, \hat{Z}_i \hat{X}_i\}_{i=1}^n$
 - 2: Build $\hat{T}, \hat{\eta}$
 - 3: Unsolved = True
 - 4: **while** Unsolved **do**
 - 5: Set a S_{ij} gauge
 - 6: Get $\hat{T}_{new}$ from Equation (30a)
 - 7: Solved = Equations (30b), (30d) are satisfied
 - 8: Unsolved = NOT Solved
 - 9: **end while**
 - 10: Apply Sign Theorem, Coefficient Theorem on T to get the graph state
-
-

Appendix N. Once the new $\hat{T}$ transfer tensor is provided, the next step consists of distinguishing between the coefficients belonging to the residual and the coefficients assigned to the PEs. To tackle such issue, the coefficient theorem turns out to be of paramount importance. The coefficients from the PEs report information about the $\hat{X}_i$ operators, whereas the coefficients from the residual report information about the $\hat{Z}_i$ operators. From the sign theorem, it is possible to check when two interactive coefficients B_i, B_j in the transfer tensor output a minus sign, tracking the neighborhood of the measured i and j qubits. Such information reduces by far the number of required operations, as it suffices to check for all the terms involving two B_i coefficients whether a minus sign occurs. Thanks to the coefficient and the sign theorems, a set of graphic rules is derived, thanks to which the residual can be mapped into

a graph and vice versa. We refer to such rules as *analysis of the residual*, and practical examples are illustrated in Appendices I and K. An overall perspective on how to build a graph state from the residual, thanks to the gauge transformation by the tensor S , is reported in Algorithm 1. The fully symmetric gauge proves to reduce the size of the graph state, with respect to the MCalculus, to 50% when applied to compiling the QFT, and to 25% for the QAOA. The results are given in Table II.

Another compiling method we include in this benchmark is provided by GRAPHIX, an open-source PYTHON library which leverages classical preprocessing of Pauli measurements and local Clifford operations in MBQC (LC-MBQC) to optimize the amount of operations and resource qubits involved when installing the graph state [33]. As it can be seen, the asymptotic performances for the

TABLE II. Number of involved qubits and entangling operations both in the MCalculus and fully symmetric gauge (FSG) by scaling with the number of input qubits n and, for the QAOA, with the layer depth p . Cyclic refers to the QAOA for tackling the MaxCut problem on cyclic graphs, whereas complete means the same algorithm is employed to solve the MaxCut for the complete graphs. Here p stands for the number of layers set in the QAOA.

	MCalculus	LC-MBQC	FSG (this work)
Resource qubits	<i>analytic complexity</i>	<i>empirical trend</i>	<i>analytic complexity</i>
QFT	$3n^2 + 2n$	$1 - \frac{3}{2}n + \frac{3}{2}n^2$	$\frac{3}{2}n^2 + \frac{3}{2}n$
Cyclic	$n(1 + 7p)$	$n + 2np$	$2n(1 + p)$
Complete	$2pn^2 + n(1 + p)$	$n + \frac{p}{2}(n + n^2)$	$\frac{1}{2}pn^2 + n\left(\frac{3}{2}p - 2\right)$
Entanglement	<i>analytic complexity</i>	<i>empirical trend</i>	<i>empirical trend</i>
QFT	$4n(n - 3/4) + 7\lceil n/2 \rceil$	$2 - 4n + 2n^2$	$4n(n - 1) + 2$
Cyclic	$n + 10np$	$3np + np^2$	$O(np^2)$
Complete	$n + pn(4n - 2)$	$O(n^2p^2)$	$O(n^3 + pn^2)$

resource qubits required are equal to those from the fully symmetric gauge. Instead, when dealing with the number of entangling operations, for the QFT GRAPHIX halves the asymptotic number of these gates ($2n^2$ instead of $4n^2$), but when considering the QAOA for complete graphs, the number of operations scales as $O(n^2p^2)$, whereas the MCalculus and the fully symmetric gauge do not exceed pn^2 or n^3 .

As for the number of controlled-phase gates to apply, for this specific gauge, we compare the analysis from MCalculus with numerical experiments. While for QFT the amount of entangling gates scales with equal trends, for the QAOA different scalings occur. If the number of layers p , i.e., the times the logic operations are repeated throughout the circuit, scales as n , or in general n^t with $t \geq 1$, the fully symmetric gauge slightly outperforms the MCalculus. In contrast, for $t < 1$, MCalculus proves to be more efficient. Some considerations arise, mainly depending on the hardware we are running the algorithm on. For instance, trapped ions demonstrate an ability to provide excellent connectivity, whereas IBM superconducting qubits are usually displaced in a honeycomb topology, which is sparser by far [55]. Summarizing all the information, raising the number of entangling operations in the graph state may hinder the computation, especially for architectures with a sparse connectivity (such as IBM). Nevertheless, for highly connected topologies such as trapped ions, this hurdle may be reduced to a polynomial law of entangling gates to be applied, while still allowing the number of involved qubits to be reduced. Moreover, we have seen that the global amount of entangling gates depends on the algorithm, on the instance of the problem, and, moreover, different gauge choices may affect the final result. Further details are given in Appendix O.

B. General considerations

We have introduced a new formalism, based on the PE of a qubit and the residual operator, to describe the graph state and to perform measurement-based computation. Next, we introduced a set of algebraic tools, such as the circuit endomorphism and the transfer tensor, to define an equivalence class between different graph states implementing the same quantum circuit. The map between the elements of the equivalence class give rise to a gauge transformation between different graph states. Thereafter, we set the action of the measurement in the new formalism, by defining the residual operator and stating the coefficient and the sign theorems, which allow us to reconstruct the graph state starting from the residual: a technique we call *analysis of the residual*. In the later sections, we fix the global picture of this work by writing the core equations of this scheme of MBQC compiling.

Thanks to such a system of equations, different ways to encode a graph state from a single unitary are now available. In order to make such equations work properly, the unitary gate must be decomposed in terms of Pauli gates. Methods such as KAK, or even gate-based circuits, provide a suitable decomposition for any unitary operator. A system of equations turns out to be more adaptive than an algorithm, providing specific solutions for any instance of unitary. Furthermore, deploying the techniques from the statements in the coefficient theorem, the sign theorem, and the following corollary, the compiling challenge can be tackled from a topological rather than a numerical perspective, avoiding memory bounds due to the allocation of n -qubits matrices with exponential size 2^n , handling instead a $G[V; E]$ graph. Future development could integrate this gauge theory with already existing compiling techniques, such as ZX diagrams or stabilizer tableaus, to integrate different approaches. The actual work provides a useful and universal example of gauge transformation in Appendix K, nominally the fully symmetric gauge. In this example, and more specifically in Appendix K, the construction of the gauge is built via a purely theoretical approach, avoiding numerical computation which may incur into any dimensional curse. Potentially, future gauges could be prospected to be found following such an approach. As already stated in Sec. II C, the analysis of the residual do require a complexity equal to $O(n^2)$, which is pretty affordable for any classical device.

Future works may focus on developing numerical and topological techniques to find the most suitable S gauge tensors for a specific unitary to be mapped into a graph state, or rather discover specific symmetries, related to certain classes of circuits or unitaries, to exploit to tackle new gauges via a theoretical approach.

IV. METHODS

A. Isomorphism between the Pauli and the canonical basis

As stated from Eq. (9), the M map represents the isomorphism between the canonical $\mathcal{C}$ and the Pauli $\mathcal{P}$ basis:

$$M = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \end{pmatrix}, \quad M : \mathcal{P} \rightarrow \mathcal{C}, \quad (31a)$$

$$M^{-1} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{pmatrix}, \quad M^{-1} : \mathcal{C} \rightarrow \mathcal{P}. \quad (31b)$$

In a tensor formalism, the M isomorphism acts on each index:

$$\eta_i = \eta_{i_1 \dots i_n} = M_{j_1 k_1} M_{j_2 k_2} \dots M_{j_n k_n} T_{k_1 \dots k_n} = M_{j_k} T_k. \quad (32)$$

B. Group symmetries on the transfer tensor

In order to generalize to n the group operations from Table I, we first focus on the 1-qubit case:

$$AaBb = AbBa \Rightarrow T_0 T_3 = T_1 T_2. \quad (33)$$

We then relabel such coefficients as C_i for the A, B and c_i for a, b , making the previous equation into

$$T_i T_j = C_i c_j C_j c_i = C_i c_j C_j c_i = T_{i \circ j} T_{j \circ i}, \quad (34)$$

where $C_0 = C_1 = A$, $C_2 = C_3 = B$, $c_0 = c_2 = a$, and $c_1 = c_3 = b$. Such a method may be adapted for the n -qubit system in the following formalism:

$$\begin{aligned} T_i T_j &= T_{i_1, i_2, \dots, i_n} T_{j_1, j_2, \dots, j_n} \\ &= C_{i_1, i_2, \dots, i_n} c_{i_1} c_{i_2} \dots c_{i_n} C_{j_1, j_2, \dots, j_n} c_{j_1} c_{j_2} \dots c_{j_n}. \end{aligned} \quad (35)$$

In the previous notation, the capital letter C_i coefficients refer to the coefficients from the residual in Eq. (28), whereas the lowercase c_{i_k} refer to the idle or active coefficients a_k, b_k of the PE $\hat{G}_k$. As all of the coefficients c_{i_k} of the PEs may be exchanged as in Eq. (33), the composition translates into the expression from Eq. (29):

$$T_i T_j = T_{i \circ j} T_{j \circ i} = T_{(i_1, i_2, \dots, i_n) \circ (j_1, j_2, \dots, j_n)} T_{(j_1, j_2, \dots, j_n) \circ (i_1, i_2, \dots, i_n)}, \quad (36)$$

which involves all the possible compositions between $(i_1, i_2, \dots, i_n) \circ (j_1, j_2, \dots, j_n)$ and vice versa between $(j_1, j_2, \dots, j_n) \circ (i_1, i_2, \dots, i_n)$.

ACKNOWLEDGMENTS

S.C. thanks Leonardo SPA for supporting his research via a PhD grant. Both S.C. and E.P have elaborated on and developed the method. They both wrote and reviewed the manuscript.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available. The data are available from the authors upon reasonable request.

APPENDIX A: NONUNITARITY OF THE PAULI ENDOMORPHISMS

lemma. The PE $\hat{G}$ is not unitary in general.

Proof. The product between a PE $\hat{G}$ and its adjoint turns out to be

$$\hat{G}_i^\dagger \hat{G}_i = |a|^2 + |b|^2 + (a^* b + b^* a) \hat{K}_i = 1 + (a^* b + b^* a) \hat{K}_i. \quad (A1)$$

Thus, the condition for being unitary holds if and only if $(a^* b + b^* a) = 0$. ■

Nevertheless, the expectation of $\hat{G}_i$ over the zero state $|0\rangle^{\otimes n}$ is equal to one, as the expectation value of $\hat{K}_i = \hat{X}_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j$ over $|0\rangle^{\otimes n}$ turns out to be null:

$${}^{\otimes n} \langle 0 | \hat{G}_i^\dagger \hat{G}_i | 0 \rangle^{\otimes n} = 1. \quad (A2)$$

Such condition is weaker than requiring $\hat{G} \in U(2^n)$. Nevertheless, if a single qubit i can be described by a PE $\hat{G}_i$ as in Eq. (B2), the condition of normalization is still satisfied.

APPENDIX B: CLOSURE OF THE STABILIZER GROUP UNDER CLIFFORD OPERATORS

lemma. If two Pauli operators $\hat{K}_1, \hat{K}_2$ commute, the new elements $\hat{K}'_1, \hat{K}'_2$ still commute. In other words, the stabilizer group is closed under the action of the Clifford algebra.

Proof. We have

$$[\hat{K}_1; \hat{K}_2] = 0 \Rightarrow 0 = \hat{U}[\hat{K}_1; \hat{K}_2] \hat{U}^\dagger = [\hat{K}'_1; \hat{K}'_2], \quad (B1)$$

where $\hat{U}$ is any element from the Clifford algebra. ■

Corollary. It is possible to entangle a register of n qubits, represented by an opportune set of n corresponding PEs, by applying a set of $\hat{C}Z_{ij}$ gates on the overall state, without affecting the commutativity of the PEs. The new state $|\Psi'\rangle$, in terms of PEs, now reads

$$\begin{aligned} |\Psi\rangle &= \prod_{i=1}^n [a_i + b_i \hat{X}_i] |0\rangle^{\otimes n} \rightarrow |\Psi'\rangle \\ &= \prod_{i=1}^n [a_i + b_i \hat{K}_i] |0\rangle^{\otimes n} = \prod_{i=1}^n \hat{G}_i(a_i, b_i) |0\rangle^{\otimes n}. \end{aligned} \quad (B2)$$

APPENDIX C: ACTION OF THE CLIFFORD ALGEBRA

Let us recall that the overall entanglement between qubits can be described by a graph, picturing the qubits as the vertices of such graph and the entanglement as its edges. The corresponding graph state, composed for instance by two qubits 1 and 2, in terms of stabilizers will be the simultaneous eigenstate of the $\langle \hat{X}_1 \hat{Z}_2; \hat{Z}_1 \hat{X}_2 \rangle$ stabilizers. While the $\hat{X}$ operators denote the nodes of the graph

or, equivalently, the qubits of the system, the $\hat{Z}$ operators refer to its edges, corresponding to the physical entanglement between the qubits [12]. Any stabilizer state can be described by a graph, the $\hat{X}_i$ corresponding to its vertices and the $\hat{Z}_j$ to the edges. Therefore, the stabilizer states are often referred to as graph states [6,10]. The purpose of this appendix is to compute the action of the elements of the Clifford group on the PEs $\hat{G}_i$.

The stabilizer group $\mathcal{S}_n$ is defined as the abelian subgroup of the Pauli group $\mathcal{P}_n$:

$$\mathcal{S}_n = \{\hat{S}_i \in \mathcal{P}_n \mid \hat{S}_i^\dagger = \hat{S}_i, [\hat{S}_i, \hat{S}_j] = 0 \forall i, j\}. \quad (\text{C1})$$

A stabilizer state $|\Psi\rangle$ can be described as a simultaneous eigenstate for such a set of commutative operators [1,56–58]. In order to build the entangled states, single- and two-qubit unitary operators are required, more specifically the Hadamard $\hat{H}$ gate and the controlled-phase $\hat{CZ}$ gate. Such operators belong to the so-called Clifford group $\mathcal{C}_n$, which features the property to map any element of the Pauli group into itself [41,57,59,60]:

$$\mathcal{C}_n = \{\hat{C} \in SU(2^n) \mid \hat{P} \in \mathcal{P}_n, \hat{C}^\dagger \hat{P} \hat{C} \in \mathcal{P}_n\}. \quad (\text{C2})$$

Some algebraic rules are summed up [22,23,61]:

$$\begin{cases} \hat{H}_i^{-1} = \hat{H}_i, \\ \hat{H}_i \hat{X}_i = \hat{Z}_i \hat{H}_i, \\ \hat{CZ}_{ij} \hat{X}_i = \hat{X}_i \hat{Z}_j \hat{CZ}_{ij}, \\ \hat{CZ}_{ij} \hat{X}_j = \hat{H}_j \hat{CZ}_{ij} \hat{H}_j, \end{cases} \quad (\text{C3})$$

with j being the target qubit and i the control qubit. Therefore, starting with a graph of unconnected vertices and making $\hat{CZ}_{ij}$ promote the connections between them, the overall entanglement of the system, i.e., the number of required entangling operations to perform, can be described by a graph and its adjacency matrix. Conversely, any regular graph can be represented by the stabilizer group, the $\hat{X}_i$ operators in Eq. (C3) pointing to the i th nodes of the graph and the $\hat{Z}_j$ referring to the (i, j) edges. It is trivial to compute that both the $\hat{C}\hat{X}$ and $\hat{C}\hat{Z}$ operators do not affect the $|0\rangle^{\otimes n}$ state.

APPENDIX D: DISRUPTIVE MEASUREMENTS IN THE HEISENBERG PICTURE

In the literature, the measurements with respect to the Pauli observables can be performed over three planes, XY , XZ , and YZ [7,53,54], each being a linear combination of two Pauli matrices, whose coefficients are given as $\cos(\theta)$, $\sin(\theta)$. In the formalism of the PEs, such measurements

can be respectively expressed as

$$\begin{aligned} \langle XY(\theta) | &= \langle 0 | [\hat{I} + e^{-i\theta} \hat{X}] \hat{Z}^s, \\ \langle YZ(\theta) | &= \langle 0 | \left[\cos\left(\frac{\theta}{2}\right) - i \sin\left(\frac{\theta}{2}\right) \hat{X} \right] \hat{X}^s, \\ \langle XZ(\theta) | &= \langle 0 | \left[\cos\left(\frac{\theta}{2}\right) + i \sin\left(\frac{\theta}{2}\right) \hat{X} \right] \hat{X}^s \hat{Z}^s. \end{aligned} \quad (\text{D1})$$

As the outcome from the measurement is probabilistic, in order to make the computation deterministic some corrections need to be fixed. Such adjustment in the MBQC literature are called byproducts, and are implemented as Pauli operators $\hat{X}$, $\hat{Z}$. A practical way to describe the overall measurement is therefore to exponentiate such byproducts to $s = 0, 1$, depending on the outcome from the measurement. In the XY plane, for instance, the observable $\hat{O}(\theta)$ from which to measure is given by

$$\hat{O}(\theta) = \cos(\theta) \hat{X} + \hat{Y} \sin(\theta) = \begin{pmatrix} 0 & e^{i\theta} \\ e^{-i\theta} & 0 \end{pmatrix}. \quad (\text{D2})$$

The eigenvalues can be evaluated from the characteristic polynomial:

$$\lambda^2 - 1 = 0 \Rightarrow \lambda_{\pm} = \pm 1. \quad (\text{D3})$$

Thus, the eigenvalue equation provides the following equations:

$$\begin{pmatrix} 0 & e^{i\theta} \\ e^{-i\theta} & 0 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} be^{i\theta} \\ ae^{-i\theta} \end{pmatrix} \stackrel{!}{=} \pm \begin{pmatrix} a \\ b \end{pmatrix}, \quad (\text{D4})$$

whose solutions are given by $b = \pm ae^{-i\theta}$, $|a|^2 + |b|^2 = 1$, and therefore the state vectors are given by

$$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ \pm e^{-i\theta} \end{pmatrix} = \frac{1}{\sqrt{2}} [\hat{I} \pm e^{-i\theta} \hat{X}] |0\rangle. \quad (\text{D5})$$

Therefore, linear combinations of multiple observables allow us to introduce rotations, when measuring the graph state. In addition, it is possible to prove that such rotation on the XY plane is equivalent to a z rotation over the $|+\rangle$ state. Introducing a global phase $e^{i\theta/2}$, the same state vector evolves into

$$\begin{aligned} & \frac{1}{\sqrt{2}} [e^{i(\theta/2)} \pm e^{-i(\theta/2)} \hat{X}] |0\rangle \\ &= \frac{1}{\sqrt{2}} \left[\hat{R}_z\left(\frac{\theta}{2}\right) |0\rangle \pm \hat{R}_z\left(\frac{\theta}{2}\right) |1\rangle \right], \end{aligned} \quad (\text{D6})$$

which can be easily rearranged into $\hat{R}_z(\theta/2) |\pm\rangle$.

As quantum measurements are not deterministic, when dealing with qubits in a $\mathbb{C}^2$ Hilbert space, the possible outcomes turn out to be 2. Quantum measurements make the

qubit collapse into a bit state, i.e., destroying any superposition, without specifying, however, which bit state they will collapse into. Therefore, when performing MBQC algorithms, a deterministic computation is required. For this reason, some correction operators, called byproducts, are introduced whenever the measurements happen. The value of such operators depends on the outcome s of the measurements, as it can be seen from Eqs. (D1). However, corrections $\hat{X}_i^s, \hat{Z}_i^s$ must be applied on the unmeasured qubits.

When performing a measurement over the i th qubit on the YZ plane, the transformation on such a qubit is implemented as follows:

$$\langle 0_i | [C + iS\hat{X}_i] \hat{X}_i^s \left[\alpha_i + \beta_i \hat{X}_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k \right] | 0_i \rangle. \quad (\text{D7})$$

Erasing the $\hat{X}_i$ operators, whose expectation value over $|0\rangle$ is null, for $s = 0, s = 1$ the possible results are

$$C\alpha_i + iS\beta_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k, \quad iS\alpha_i + C\beta_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k$$

Thus, a statement follows: if the measured qubit is set on a superposition state $|+\rangle$, or in any state with $\alpha \neq 0$ or $\beta \neq 0$, the byproduct correction consists of applying the $\hat{Z}$ neighbors of the qubit,

$$\left[\bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k \right]^s \langle 0_i | [C + iS\hat{X}_i] \hat{X}_i^s \left[\alpha_i + \beta_i \hat{X}_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k \right] | 0_i \rangle. \quad (\text{D8})$$

When performing a measurement over the i th qubit on the XY plane, the transformation on such qubit is implemented as follows:

$$\langle 0_i | [\hat{I} + e^{-i\theta} \hat{X}_i] \hat{Z}_i^s \left[\alpha_i + \beta_i \hat{X}_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k \right] | 0_i \rangle. \quad (\text{D9})$$

Erasing the $\hat{X}_i$ operators, whose expectation value over $|0\rangle$ is null, for $s = 0, s = 1$ the possible results are

$$\alpha_i \pm e^{-i\theta} \beta_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k. \quad (\text{D10})$$

If any of the neighbors of the i th qubit is set in a superposition state $|+\rangle$, or in a more generic formulation $\alpha \neq 0, \beta \neq 0$, it is possible to apply the $\hat{K}_l = \hat{X}_l \otimes_{j \in \mathcal{N}(l)} \hat{Z}_j$ stabilizer, with $l \in \mathcal{N}(k)$ (i.e., belonging to the neighborhood of k) in order to correct the measurement:

$$[\hat{K}_l]^s \left[\alpha_i \pm e^{-i\theta} \beta_i \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k \right] [\alpha_l + e^{i\gamma} \alpha_l \hat{K}_l]. \quad (\text{D11})$$

APPENDIX E: EVOLUTION OF THE RESIDUAL OPERATOR

Proof. To compute how a generic measurement $\langle \gamma_i |$ affects the overall state in the formalism of the PEs, first commute the i th PE up to the leftmost position. Such operation is always allowed thanks to the commutativity of the stabilizer group:

$$|\Psi\rangle = [a_i + b_i \hat{K}_i] \prod_{\substack{j=1 \\ j \neq i}}^n \hat{G}_j(a_j, b_j) |0\rangle^{\otimes n}. \quad (\text{E1})$$

Thereafter, contract the $|0_i\rangle$ state to the i th PE, which has been granted to be possible as stated in Eq. (18):

$$[a_i + b_i \hat{K}_i] |0_i\rangle \prod_{\substack{j=1 \\ j \neq i}}^n \hat{G}'_j(a_j, b_j) |0\rangle^{\otimes n-1}, \quad (\text{E2})$$

where G' are the PEs affected by such operation, as described in the previous lemma. In second place, apply the $\langle \gamma_i | = \langle 0_i | [\alpha_i^* + \beta_i^* \hat{X}_i]$ measurement over the i th PE. The residual is yielded by the following expectation value over the i th qubit:

$$\begin{aligned} \hat{R} &= \langle 0_i | [\alpha_i^* + \beta_i^* \hat{X}_i] \left[a_i + b_i \hat{X}_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j \right] | 0_i \rangle \\ &= \alpha_i^* a_i + \beta_i^* b_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j = A_i + B_i \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j, \end{aligned} \quad (\text{E3})$$

which matches Eq. (19). Here, $\mathcal{N}(i)$ is again the set of qubits entangled with the i th qubit before its measurement, and $A_i = \alpha_i^* a_i, B_i = \beta_i^* b_i$. No $\hat{X}_i$ terms appear anymore, as $\langle 0 | \hat{X} | 0 \rangle = 0$. The residual operator $\hat{R}$ can be settled as the operator which acts on the unmeasured qubits, as a consequence of the entanglement with the previously measured ones. The new $|\Psi'\rangle$ state, after such measurement, now reads as

$$|\Psi'\rangle = \hat{R} \prod_{\substack{j=1 \\ j \neq i}}^n \hat{G}'_j(a_j, b_j) |0\rangle^{\otimes n-1}. \quad (\text{E4})$$

To complete the proof, one recursively applies the measurement operation $\langle \gamma_l |$ over the $|\Psi'\rangle$ state, as follows:

$$\begin{aligned} \langle \gamma_l | \Psi' \rangle &= \langle 0_l | [\alpha_l^* + \beta_l^* \hat{X}_l] \hat{R} [a_l + b_l \hat{K}_l] | 0_l \rangle \\ &\times \prod_{\substack{j=1 \\ j \neq i, l}}^n \hat{G}''_j(a_j, b_j) |0\rangle^{\otimes n-2}. \end{aligned} \quad (\text{E5})$$

As $\hat{K}_l = \hat{X}_l \otimes_{f \in \mathcal{N}(l)} \hat{Z}_f$ and $\hat{R} = A_i + B_i \otimes_{j \in \mathcal{N}(i)} \hat{Z}_j$, the new residual reads

$$\alpha_i^* \hat{R}' a_l + \beta_i^* b_l \langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle, \quad (\text{E6})$$

where $\hat{R}' = A_i + B_i \otimes_{j \in \mathcal{N}(i) \setminus \{l\}} \hat{Z}_j$, as $\langle 0_l | \hat{Z}_l | 0_l \rangle = 1$. From the new operator $\langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle$ no $\hat{X}_l$ term survives, therefore the new residual is still not composed by any $\hat{X}$ operator, as occurred during the first iteration in Eq. (19). It is possible to relabel the new residual as $\hat{R}$:

$$\hat{R} \leftarrow \alpha_i^* \hat{R}' a_l + \beta_i^* b_l \langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle \quad (\text{E7})$$

and iterate such procedure recursively for any measurement. ■

APPENDIX F: PROOF OF THE COEFFICIENT THEOREM

Proof of the coefficient theorem. Recalling Eq. (E7), it follows that

$$\hat{R} \leftarrow A_l \hat{R}' + B_l \langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle. \quad (\text{F1})$$

Since this operation is repeated recursively, it holds for any measurement. ■

APPENDIX G: PROOF OF THE SIGN THEOREM

Proof of the sign theorem. The proof follows from Eqs. (E6) and (E7). The second term from Eq. (E6) can be rearranged as

$$\begin{aligned} B_l \langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle \\ = A_i B_l \langle 0_l | \hat{X}_l \hat{K}_l | 0_l \rangle + B_i B_l \langle 0_l | \hat{X}_l \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j \hat{K}_l | 0_l \rangle. \end{aligned} \quad (\text{G1})$$

The $A_i B_l$ belong to the first order of recursion, as in Eq. (19):

$$A_i B_l \langle 0_l | \hat{X}_l \hat{K}_l | 0_l \rangle = A_i B_l \bigotimes_{w \in \mathcal{N}(l)} \hat{Z}_w, \quad (\text{G2})$$

whereas the second term reads

$$\begin{aligned} B_i B_l \langle 0_l | \hat{X}_l \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j \hat{K}_l | 0_l \rangle \\ = (-)^s B_i B_l \langle 0_l | \bigotimes_{j \in \mathcal{N}(i)} \hat{Z}_j \hat{X}_l \hat{K}_l | 0_l \rangle \\ = (-)^s B_i B_l \bigotimes_{j \in \mathcal{N}(i) \setminus \{l\}} \hat{Z}_j \bigotimes_{w \in \mathcal{N}(l)} \hat{Z}_w, \end{aligned} \quad (\text{G3})$$

where $s \in \{0, 1\}$ represents the condition $l \in \mathcal{N}(i)$. If it holds, then the $\hat{X}_l \hat{Z}_l$ operators anticommute, yielding a

minus sign and thus $s = 1$; otherwise, if no $\hat{Z}_l$ is present (because $l \notin \mathcal{N}(i)$), then $s = 0$. The final expression reads

$$\begin{aligned} B_l \langle 0_l | \hat{X}_l \hat{R} \hat{K}_l | 0_l \rangle = A_i B_l \bigotimes_{w \in \mathcal{N}(l)} \hat{Z}_w + (-)^s B_i B_l \\ \bigotimes_{j \in \mathcal{N}(i) \setminus \{l\}} \hat{Z}_j \bigotimes_{w \in \mathcal{N}(l)} \hat{Z}_w. \end{aligned} \quad (\text{G4})$$

The two tensor products of the term on the right-hand side can be decomposed as follows:

$$\bigotimes_{j \in \mathcal{N}(i) \cap \mathcal{N}(l) \setminus \{l\}} \hat{Z}_j^2 \bigotimes_{w \in \mathcal{N}(l) \cup \mathcal{N}(i) \setminus \mathcal{N}(l) \cap \mathcal{N}(i) \setminus \{l\}} \hat{Z}_w, \quad (\text{G5})$$

from which the assumption of the theorem, setting $\{l\} = \mathcal{M}$. A practical example is provided in Fig. 4. As such procedure can be iterated, it holds for any measurement. ■

APPENDIX H: PROOF OF THE OPERATOR COROLLARY

Proof of the operator corollary. As the idle coefficients are not associated with any operator, recalling Eq. (G2) the interactive terms B_i in the residual display a $\hat{Z}_k$ for the output qubits $k \in \mathcal{N}(i)$:

$$A_1 \cdots A_{i-1} B_i A_{i+1} \cdots A_n \bigotimes_{k \in \mathcal{N}(i)} \hat{Z}_k, \quad (\text{H1})$$

with B_i being the sole interactive coefficient in the sequence of A_j , $j \neq i$. If more interactive terms would contribute to the sequence, the sign theorem should be invoked. ■

APPENDIX I: AN EXAMPLE OF GRAPHIC RULES

In this appendix, we deploy the case of a 5-qubit graph state built as in Fig. 4 where 4 and 5 qubits are measured, resulting in a 3-qubit output state. Such a process provides a practical example to apply the two theorems until now exposed. In order to assess how the entanglement between measured qubits affects the minus sign as stated by the sign theorem, we compare two cases where the 4 and 5 qubits may be entangled or not. In the first place, we write down the PEs of the system, following the Clifford rules of the entangling gates as expressed in Eq. (B2):

$$\begin{aligned} |\Psi_5\rangle = [a_1 + b_1 \hat{X}_1 \hat{Z}_4 \hat{Z}_5] [a_2 + b_2 \hat{X}_2 \hat{Z}_4] [a_3 + b_3 \hat{X}_3 \hat{Z}_5] \\ \times [a_4 + b_4 \hat{X}_4 \hat{Z}_2 \hat{Z}_1 \hat{Z}_5^s] \cdot [a_5 + b_5 \hat{X}_5 \hat{Z}_1 \hat{Z}_3 \hat{Z}_4^s] |0\rangle^{\otimes 5}, \end{aligned} \quad (\text{I1})$$

with $s \in \{0, 1\}$ depending on whether the entanglement occurs between 4 and 5, represented by the dashed line

in Fig. 4. We then choose to measure the qubit 4, which measurement sets the overall state $|\Psi_5\rangle \rightarrow |\Psi_4\rangle$:

$$\begin{aligned} \langle \gamma_4 | \Psi_5 \rangle &= \langle 0_4 | [\alpha_4 + \beta_4 \hat{X}_4] [a_4 + b_4 \hat{X}_4 \hat{Z}_2 \hat{Z}_1 \hat{Z}_5^s] | 0_4 \rangle \\ &[a_1 + b_1 \hat{X}_1 \hat{Z}_5] [a_2 + b_2 \hat{X}_2] [a_3 + b_3 \hat{X}_3 \hat{Z}_5] \\ &\times [a_5 + b_5 \hat{X}_5 \hat{Z}_1 \hat{Z}_3] | 0 \rangle^{\otimes 4} = \hat{R} | \Psi_4 \rangle. \end{aligned} \quad (12)$$

The expectation over $|0_4\rangle$ yields the residual of the measurement:

$$\begin{aligned} \hat{R} &= \langle 0_4 | [\alpha_4 + \beta_4 \hat{X}_4] [a_4 + b_4 \hat{X}_4 \hat{Z}_1 \hat{Z}_2 \hat{Z}_5^s] | 0_4 \rangle \\ &= \alpha_4 a_4 + \beta_4 b_4 \hat{Z}_1 \hat{Z}_2 \hat{Z}_5^s = A_4 + B_4 \hat{Z}_1 \hat{Z}_2 \hat{Z}_5^s \end{aligned} \quad (13)$$

Afterwards, we perform the measurement over the qubit 5 on the $|\Psi_4\rangle$ state:

$$\begin{aligned} \langle \gamma_5 | \hat{R} | \Psi_4 \rangle &= \langle 0_5 | [\alpha_5 + \beta_5 \hat{X}_5] \hat{R} [a_5 + b_5 \hat{X}_5 \hat{Z}_1 \hat{Z}_3] | 0_5 \rangle [a_1 + b_1 \hat{X}_1] \\ &\times [a_2 + b_2 \hat{X}_2] [a_3 + b_3 \hat{X}_3] | 0 \rangle^{\otimes 3}. \end{aligned} \quad (14)$$

We now compute the residual after the second measurement:

$$\begin{aligned} \langle 0_5 | [\alpha_5 + \beta_5 \hat{X}_5] [A_4 + B_4 \hat{Z}_1 \hat{Z}_2 \hat{Z}_5^s] [a_5 + b_5 \hat{X}_5 \hat{Z}_1 \hat{Z}_3] | 0_5 \rangle &= A_4 A_5 + A_4 B_5 \hat{Z}_1 \hat{Z}_3 + B_4 A_4 \hat{Z}_1 \hat{Z}_2 \\ &+ B_4 B_5 \langle 0_5 | \hat{X}_5 \hat{Z}_2 \hat{Z}_3 \hat{Z}_5^s \hat{X}_5 | 0_5 \rangle. \end{aligned} \quad (15)$$

Depending on $s = 0$ or $s = 1$, it holds a minus sign in front of the two interactive coefficients $B_4 B_5$. After the two measurements, the residual reads

$$\hat{R} = A_4 A_5 + A_4 B_5 \hat{Z}_1 \hat{Z}_3 + B_4 A_5 \hat{Z}_1 \hat{Z}_2 + (-)^s B_4 B_5 \hat{Z}_2 \hat{Z}_3. \quad (16)$$

As stated by the operator corollary, each interactive coefficient carries a sequence of $\hat{Z}_i$ operators, i pointing to the output qubits the measured ones were entangled with. The sign theorem, instead, states that if two measured qubits were entangled, a minus sign arises in front of their interactive coefficients. Thanks to such theorems, it is possible to achieve the result in Eq. (16) avoiding all of the algebraic calculations.

APPENDIX J: GAUGE TRANSFORMATIONS FOR TWO QUBITS

In this appendix, we adapt the multiqubit formalism for the gauge transformations to the particular case $n = 2$, in order to ease the comprehension of the tensor depiction of the MBQC compiling equations. In order to introduce transformations on systems of two qubits, we describe at

TABLE III. How to arrange, in the tensor format, the coefficients to their respective PEs from the Pauli basis, which are described as a linear combination in Eq. (J1). The transfer tensor for the single-qubit case, indeed, matches the first row (or first column) of the table.

I	X_2	Z_2	$Z_2 X_2$
X_1	$X_1 X_2$	$X_1 Z_2$	$X_1 Z_2 X_2$
Z_1	$Z_1 X_2$	$Z_1 Z_2$	$Z_1 Z_2 X_2$
$Z_1 X_1$	$Z_1 X_1 X_2$	$Z_1 X_1 Z_2$	$Z_1 X_1 Z_2 X_2$

first the circuit endomorphism for such system, assuming the output qubits not to be entangled:

$$\hat{\eta}_2 = [C_0 + C_1 \hat{Z}_1 + C_2 \hat{Z}_2 + C_{12} \hat{Z}_1 \hat{Z}_2] \prod_{j=1}^2 \hat{G}(a_j, b_j). \quad (J1)$$

The next step consists of representing the transfer tensor T . From a 1-qubit system to a 2-qubit one, $\hat{T}$ scales from a 1-rank T_i tensor to a 2-rank T_{ij} . Its expression is therefore reshaped into

$$\begin{bmatrix} C_0 a_1 a_2 & C_0 a_1 b_2 & C_2 a_1 a_2 & C_2 a_1 b_2 \\ C_0 b_1 a_2 & C_0 b_1 b_2 & C_2 b_1 a_2 & C_2 b_1 b_2 \\ C_{12} a_1 a_2 & C_{12} a_1 b_2 & C_{12} b_1 a_2 & C_{12} b_1 b_2 \\ C_{12} b_1 a_2 & C_{12} b_1 b_2 & C_{12} a_1 a_2 & C_{12} a_1 b_2 \end{bmatrix}. \quad (J2)$$

A representation of all the coefficients with respect to their PEs in the Pauli basis is given in Table III. The transformations induced by M and M^{-1} can be generalized to

$$T'_{i_1 j_1} = S_{i_1 i_2 j_1 j_2} T_{i_2 j_2} = S_{i_1 i_2 j_1 j_2} M_{i_2 i_3} M_{j_2 j_3} \eta_{i_3 j_3}. \quad (J3)$$

The following compact notation on the indices helps in lightening such formalism:

$$T'_i = S_{ij} T_j = S_{ij} M_{jk} \eta_k. \quad (J4)$$

In such case, $M_{ij} = M_{i_2 i_3} M_{j_2 j_3}$, whereas the gauge transformation S_{ij} is not constrained to factorize into two 2-rank tensors.

The group operations for the symmetries over the transfer tensor T need to be explained for the case $n = 2$. In the first place, we may review the group operations as swaps between the coefficients of the PE:

$$AaBb = AbBa \Rightarrow T_0 T_3 = T_1 T_2. \quad (J5)$$

We may relabel such coefficients as C_i for the A , B and c_i for a , b , making the previous equation into

$$T_i T_j = C_i c_i C_j c_j = C_i c_j C_j c_i = T_{i \circ j} T_{j \circ i}, \quad (J6)$$

where $C_0 = C_1 = A$, $C_2 = C_3 = B$, $c_0 = c_2 = a$, and $c_1 = c_3 = b$. Such method may be adapted for the 2-qubit

system in the following fashion:

$$T_{i_1 j_1} = C_{i_1 j_1} c_{i_1} c_{j_1}, \quad (\text{J7})$$

where, referring to the tensor format in Eq. (J2), $C_{0,0} = C_{0,1} = C_{1,0} = C_{1,1} = C_0$, then $C_{2,0} = C_{2,1} = C_{3,0} = C_{3,1} = C_1$ and so on. Indeed, the above expression can be read as an identity:

$$T_{i_1 j_1} = C_{i_1 j_1} c_{i_1} c_{j_1} = T_{i_1 \circ i_1 j_1 \circ j_1}, \quad (\text{J8})$$

which composition returns, in fact, the identity. Such consideration nonetheless leads to a nontrivial result:

$$\begin{aligned} T_{i_1 j_1} T_{i_2 j_2} &= C_{i_1 j_1} c_{i_1} c_{j_1} C_{i_2 j_2} c_{i_2} c_{j_2} \\ &= C_{i_1 j_1} c_{i_2} c_{j_1} C_{i_2 j_2} c_{i_1} c_{j_2} = T_{i_1 \circ i_2 j_1} T_{i_2 \circ i_1 j_2} \end{aligned} \quad (\text{J9})$$

and the same operation may be repeated for j_1 and j_2 . Note that this formalism on the indices of the transfer tensor has been justified in Sec. IV B. Group compositions between different indices (e.g., i_k and j_k) commute each other. Such freedom can be encoded in the following formalism:

$$T_i T_j = T_{i \circ j} T_{j \circ i} = T_{(i_1 j_1) \circ (i_2 j_2)} T_{(i_2 j_2) \circ (i_1 j_1)}, \quad (\text{J10})$$

which expression involves all the possible compositions between $(i_1, j_1) \circ (i_2, j_2)$ and vice versa, including the trivial ones (the identity):

$$\begin{cases} T_{i_1 \circ i_1 j_1 \circ j_1} T_{i_2 \circ i_2 j_2 \circ j_2}, \\ T_{i_1 \circ i_2 j_1} T_{i_2 \circ i_1 j_2}, \\ T_{i_1 j_1 \circ j_2} T_{i_2 j_2 \circ j_1}, \\ T_{i_1 \circ i_2 j_1 \circ j_2} T_{i_2 \circ i_1 j_2 \circ j_1}. \end{cases} \quad (\text{J11})$$

APPENDIX K: THE FULLY SYMMETRIC GAUGE

In this appendix, we introduce the fully symmetric gauge S' , a universal gauge which fits for any unitary transformation. For the single-qubit case, the gauge operator reads as

$$S' = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{pmatrix}. \quad (\text{K1})$$

It is straightforward to compute $\det(S') = 0$, which makes such transformation a singular one. We now briefly prove how such $1d$ transformation can be achieved. The action of the circuit endomorphism, for the $n = 1$ qubit case, can

be explicitly written as

$$\hat{\eta} |0\rangle = \begin{pmatrix} \eta_{00} & \eta_{01} \\ \eta_{10} & \eta_{11} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \quad (\text{K2})$$

Recall the isomorphism M in Eq. (31) between the canonical and the Pauli bases:

$$\begin{aligned} M &= \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \end{pmatrix}, \quad M : \mathcal{P} \rightarrow \mathcal{C}, \\ M^{-1} &= \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{pmatrix}, \quad M^{-1} : \mathcal{C} \rightarrow \mathcal{P}. \end{aligned} \quad (\text{K3})$$

It is possible to set the following choice of gauge when building the transfer matrix, and thus the graph state:

$$\begin{cases} T_{00} \rightarrow T_{00} + \gamma = \tilde{T}_{00}, \\ T_{01} \rightarrow T_{01} + \gamma' = \tilde{T}_{01}, \\ T_{10} \rightarrow T_{10} - \gamma = \tilde{T}_{10}, \\ T_{11} \rightarrow T_{11} + \gamma' = \tilde{T}_{11}, \end{cases} \quad (\text{K4})$$

with γ , γ' being two constants, and $\tilde{T}$ the new transfer matrix. In fact, such operation makes the $\tilde{\eta}$ invariant, according to Eqs. (31) in fact $\eta_{00} = T_{00} + \gamma + T_{10} - \gamma$ and $\eta_{10} = T_{01} + \gamma' - (T_{11} + \gamma')$. Thus, the relation $\tilde{\eta}' = \tilde{\eta}$ is satisfied. Imposing the symmetries on the new transfer tensor $\tilde{T}$, it is possible to get the values for γ and γ' :

$$\begin{cases} T'_{00} = T'_{01} \\ T'_{10} = T'_{11} \end{cases} \Rightarrow \begin{cases} T_{00} + \gamma = T_{01} + \gamma', \\ T_{10} - \gamma = T_{11} + \gamma', \end{cases} \quad (\text{K5})$$

which always admits a solution,

$$\begin{cases} \gamma' = \frac{1}{2}(T_{00} + T_{10} - T_{01} - T_{11}), \\ \gamma = \frac{1}{2}(T_{01} + T_{10} - T_{00} - T_{11}). \end{cases} \quad (\text{K6})$$

Of course, when the transfer matrix is already symmetric, the γ and γ' vanish, as far as $T_{00} = T_{01}$ and $T_{10} = T_{11}$. The new transfer matrix $\tilde{T}$ thus reads as

$$T' = \frac{1}{2} \begin{bmatrix} T_{00} + T_{01} + T_{10} - T_{11} & T_{00} + T_{01} + T_{10} - T_{11} \\ T_{00} - T_{01} + T_{10} + T_{11} & T_{00} - T_{01} + T_{10} + T_{11} \end{bmatrix}, \quad (\text{K7})$$

with T_{ij} being the coefficient of the original transfer matrix T . It is quite straightforward to show that T' now satisfies the constraint of symmetry $T_i T_j = T_{i \circ j} T_{j \circ i}$. The same

transformation can be better represented in the tensor format:

$$T'_i = S_{ij} T_j, \quad (\text{K8a})$$

$$\begin{pmatrix} T'_{00} \\ T'_{01} \\ T'_{10} \\ T'_{11} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} T_{00} \\ T_{01} \\ T_{10} \\ T_{11} \end{pmatrix}. \quad (\text{K8b})$$

Equation (K8a) indeed matches the cardinal equation for a single-qubit system

$$T'_i = S_{ij} T_j = S_{ij} M_{jk} \eta_k. \quad (\text{K9})$$

The same transformation for the 1-qubit case can be described via the cardinal equation

$$\eta'_i = M_{ij}^{-1} T'_j = M_{ij}^{-1} S_{jk} T_k. \quad (\text{K10})$$

Here $\eta_0 = \eta_{00}$ and $\eta_2 = \eta_{10}$, which are the elements belonging to the vector circuit we want to preserve. Indeed, in order to check this fact, a manipulation of the previous equation turns out to be useful:

$$\eta'_i = S'_{ik} T_k, \quad (\text{K11})$$

where $S'_{ik} = M_{ij}^{-1} S_{ij}$. Therefore, in order to preserve $\vec{\eta}$, we want $S'_{0j} = M_{0j}^{-1}$ and $S'_{2j} = M_{2j}^{-1}$, so that the elements from the vector circuit are still preserved. By a simple computation, for the fully symmetric gauge S^f we achieve

$$\begin{aligned} M_{ij}^{-1} S'_{jk} &= \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{pmatrix} \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & -1 \end{pmatrix}. \end{aligned} \quad (\text{K12})$$

Hence, $S'_{0j} = (1, 0, 1, 0) = M_{0j}^{-1}$, $S'_{2j} = (0, 1, 0, -1) = M_{2j}^{-1}$, which accomplish our purpose. The multiqubit case is quite straightforward, as the fully symmetric gauge S_n^f is introduced as a factorized product of the single-qubit version S^f :

$$S_n^f = \bigotimes_{i=1}^n S^f. \quad (\text{K13})$$

Therefore, the previous cardinal equation can be rewritten as

$$\begin{aligned} \eta_{i_1 \dots i_n} &= S_{i_1 \dots i_n j_1 \dots j_n}^f M_{j_1 k_1}^{-1} \dots M_{j_n k_n}^{-1} T_{k_1 \dots k_n} \\ &= S_{i_1 j_1}^f \dots S_{i_n j_n}^f M_{j_1 k_1}^{-1} \dots M_{j_n k_n}^{-1} T_{k_1 \dots k_n} \\ &= S_{i_1 k_1}^f \dots S_{i_n k_n}^f T_{k_1 \dots k_n}. \end{aligned} \quad (\text{K14})$$

Nonetheless, each of the S_{ikl}^f tensors obeys to the constraint $S_{0j}^f = M_{0j}^{-1}$, proving the scalability of the S_n^f gauge along with n .

For a generic n -qubit case, the fully symmetric gauge can be set to be $S_{ij}^f = \bigotimes_{l=1}^n S_{ij}^f$. Consider now the transfer tensor for n qubits in the matrix format. First we need to introduce some considerations. In order to reduce the tensor $T_{i_1 i_2 \dots i_n}$ to a matrix, we first map the index $i_p \in \{0, 1, 2, 3\}$ to $i_p, j_p \in \{0, 1\}$. In this case, the explicit map between the two formalisms on the indices is

$$\begin{cases} 0 \rightarrow 00, \\ 1 \rightarrow 01, \\ 2 \rightarrow 10, \\ 3 \rightarrow 11. \end{cases} \quad (\text{K15})$$

The transfer tensor now reads $T_{i_1 j_1 i_2 j_2 \dots i_n j_n}$. The next step consists of folding all the i_p, j_p indices into just two by introducing the following notation:

$$T_{\sum_{l=0}^n 2^{n-l} i_{l+1}, \sum_{l=0}^n 2^{n-l} j_{l+1}}. \quad (\text{K16})$$

For instance, for $n = 3$, the coefficient $T_{i_1 j_1 i_2 j_2 i_3 j_3}$ is pointed to by the new one $T_{4i_1+2i_2+i_3, 4j_1+2j_2+j_3}$. After applying the symmetric gauge for the 1-qubit case, we have noted in Eq. (K7) that $T'_{ij} = T_{\bar{i}\bar{j}}$, where $\bar{0} = 1$ and $\bar{1} = 0$ (i.e., the negation). Since the gauge transformations S_{ij}^f act index-wise, see Eqs. (30a) and (30c), we may state that, for any p :

$$T'_{i_1 j_1 \dots i_p j_p \dots i_n j_n} = T_{i_1 \bar{j}_1 \dots i_p \bar{j}_p \dots i_n j_n}. \quad (\text{K17})$$

We now prove that all the columns are equal by employing the matrix formalism in Eq. (K16). We may apply random transformations on the j_l coefficients converting them in k_l , in other words $k_l = j_l$ or $k_l = \bar{j}_l$ at will. This freedom translates into a precise symmetry:

$$\begin{aligned} T_{\sum_{l=0}^n 2^{n-l} i_{l+1}, \sum_{l=0}^n 2^{n-l} j_{l+1}} &= T_{i_1 j_1 i_2 j_2 \dots i_n j_n} \\ &= T_{i_1 k_1 i_2 k_2 \dots i_n k_n} = T_{\sum_{l=0}^n 2^{n-l} i_{l+1}, \sum_{l=0}^n 2^{n-l} k_{l+1}}. \end{aligned} \quad (\text{K18})$$

After the fully symmetric gauge has been applied, thus all the columns are displayed equal to each other:

$$T = \frac{1}{2^n} \begin{pmatrix} T_1 & T_1 & \cdots & T_1 \\ T_2 & T_2 & \cdots & T_2 \\ \vdots & \vdots & \ddots & \vdots \\ T_N & T_N & \cdots & T_N \end{pmatrix}, \quad (\text{K19})$$

where $N = 2^n$. In this form, Eq. (30d) is already satisfied. This consideration can be proved for the single-qubit case, then it is straightforward to be generalized. Returning to the index conversion in Eq. (K15), the tensor (T_0, T_1, T_2, T_3) can be converted into

$$\begin{pmatrix} T_0 & T_1 \\ T_2 & T_3 \end{pmatrix}. \quad (\text{K20})$$

Thus, relying on the group operations stated in Table I, we can see that, for the nontrivial operations, 0 is mapped into 1 and vice versa, as well as 2 into 3 (and vice versa). If the two columns are equal, this swap leaves the matrix unaltered, proving that Eq. (30d) is satisfied.

Next, the $1/2^n$ coefficient can be factored into $2n$ factors equal to $1/\sqrt{2}$. The output qubits belonging to the PEs, whose number amounts to n , can be set as $a_i = b_i = 1/\sqrt{2}$, meaning that the output qubits lie in an overall superposition state $|+\rangle$. Nevertheless, other n terms survive, which can be addressed as supplementary coefficients A_i, B_i to the residual. A practical visualization is sketched in Fig. 5(a). Here, the qubits referred to as B_i are those corresponding to the PE $[A_i + B_i \hat{K}_i]$. Instead, the output qubits referred to as O_i are those represented by the coefficients $[a_i + b_i \hat{K}_i]$. The remaining qubits in the residual have their coefficients included in the T_i terms from the above equations. While each interactive term b_i from the PEs contributes with the corresponding $\hat{X}_i$ operator, the A_i ones from the residual,

$i \neq 0$, carry a $\hat{Z}_i$ Pauli matrix. Now the technique which we called *analysis of the residual* can be performed: fitting such coefficients into the transfer tensor in Eq. (K19), the T_{i0} matrix elements will shape into

$$\begin{aligned} T_{00} &= T_1 A_1 A_2 \cdots A_n a_1 a_2 \cdots a_n \rightarrow \hat{I}, \\ T_{10} &= T_2 B_1 A_2 \cdots A_n a_1 a_2 \cdots a_n \rightarrow \hat{X}_1, \\ T_{20} &= T_3 A_1 B_2 \cdots A_n a_1 a_2 \cdots a_n \rightarrow \hat{Z}_2, \\ &\vdots \\ T_{(N-1)0} &= T_n B_1 B_2 \cdots B_n a_1 a_2 \cdots a_n \rightarrow \hat{Z}_1 \cdots \hat{Z}_n, \end{aligned} \quad (\text{K21})$$

whereas the T_{0j} elements will read

$$\begin{aligned} T_{00} &= T_1 A_1 A_2 \cdots A_n a_1 a_2 \cdots a_n \rightarrow \hat{I}, \\ T_{01} &= T_1 A_1 A_2 \cdots A_n b_1 a_2 \cdots a_n \rightarrow \hat{X}_1, \\ T_{02} &= T_1 A_1 A_2 \cdots A_n a_1 b_2 \cdots a_n \rightarrow \hat{X}_2, \\ &\vdots \\ T_{0(N-1)} &= T_1 A_1 A_2 \cdots A_n b_1 b_2 \cdots b_n \rightarrow \hat{X}_1 \cdots \hat{X}_n. \end{aligned} \quad (\text{K22})$$

Thanks to operator corollary, the B_i interactive terms are linked to the $\hat{Z}_i$ operators. In contrast, all the coefficients contained inside the T_i terms are related to qubits not neighboring with the output ones. Referring to Fig. 5(b), the a_i, b_i output coefficients belong to the output qubit $|o\rangle$, the A_i, B_i coefficients to the $|B\rangle$ qubit, all the other coefficients in the T_i terms form the PEs for the $|a\rangle$ and $|\theta\rangle$ states.

The *analysis of the residual* returns that the combination of the A_i, B_i along with the a_j, b_j coefficients gives rise to all the $\hat{Z}_i$ and $\hat{X}_i$ terms. In order to study the interactions

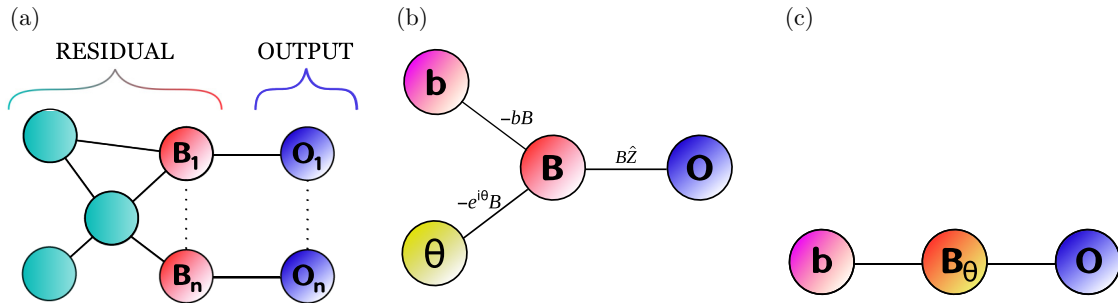


FIG. 5. (a) The fully symmetric gauge outputs a graph state where the outputs are set in a $|+\rangle$ state, linked to a second layer of qubits in the $|+\rangle$ state. From such layer, all the connections with the residual are established. (b) The graph state for a $R_x(\theta)$ rotation resulting from a fully symmetric gauge, as exposed in Appendix L. All the qubits, except the O one, have to be measured, specifically in the $\langle +|$ basis. Above the edges, the terms of the residuals with the interactive coefficients are shown. While the minus sign is justified by the sign theorem, the $\hat{Z}$ operator is justified by the operator corollary. (c) The same graph, with reduced dimension thanks to the merging of the two nodes in the residual. Thanks to such a reduction, the second qubit is measured in the XY plane instead of by $|+\rangle$. As $|o\rangle = |+\rangle$, $\alpha_o = \beta_o = 1/\sqrt{2}$, and therefore it is possible to apply the opportune byproducts.

between the qubits in the residual, without any regard to the output, focus on the T_{00} term, as the $\hat{Z}_i$ operators do not appear. In addition, in order to study the interaction between the residual and the PEs, focus on the T_{1j} elements, $j \geq 1$. From the T_{00} element, in fact, no interactive terms B_i , b_i appear, along with no $\hat{X}_i$, $\hat{Z}_i$ operators. Thus, it is possible to apply the sign theorem, focusing over all the coupled interacting terms B_k , B_l contained inside the T_1 coefficient, filtering all the other terms with a number $m \neq 2$ of B_p interacting terms. For this same reason, there is no need to allocate a $2^n \times 2^n$ matrix to perform computation, but just a very small subset of it, involving the terms needed for the analysis of the residual, i.e., $O(n^2)$ terms instead of $2^n \times 2^n$, as stated in Sec. II C. Resorting back to the sign theorem, it is possible to check which interactive coefficients from T_i outputs a minus sign in front of the B_i . Therefore, all the terms with a number $m \neq 1$ of interactive terms will be filtered away. In such conditions, if two interactive coefficients B_k , B_l yield a minus sign in front of the term, the stabilizers of the underlying graph state will reshape as

$$\langle \hat{X}_k; \hat{X}_l \rangle \rightarrow \langle \hat{X}_k \hat{Z}_l; \hat{X}_l \hat{Z}_k \rangle, \quad (\text{K23})$$

thus expressing the entanglement between the two qubits in the stabilizer formalism. A picture representing the coefficients from the $1/2^n$ factor is illustrated in Fig. 5(a). A practical example for a full symmetric gauge transformation is provided for the $\hat{R}_x$ rotations in Appendix L.

APPENDIX L: ONE-QUBIT GATE EXAMPLE

The action of the $\hat{R}_x(\theta)$ operator, in terms of circuit endomorphism, can be depicted as

$$\left[\cos\left(\frac{\theta}{2}\right) + i \sin\left(\frac{\theta}{2}\right) \hat{X} \right] [a + b \hat{X}] |0\rangle. \quad (\text{L1})$$

In the canonical basis, such operation assumes the following matrix form:

$$\begin{pmatrix} a \cos\left(\frac{\theta}{2}\right) + ib \sin\left(\frac{\theta}{2}\right) & b \cos\left(\frac{\theta}{2}\right) + ia \sin\left(\frac{\theta}{2}\right) \\ b \cos\left(\frac{\theta}{2}\right) + ia \sin\left(\frac{\theta}{2}\right) & a \cos\left(\frac{\theta}{2}\right) + ib \sin\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \quad (\text{L2})$$

For the sake of simplicity, we now rename $C = \cos(\theta/2)$ and $S = \sin(\theta/2)$. The transfer matrix reads as

$$T = \begin{pmatrix} Ca + iSb & Cb + iSa \\ 0 & 0 \end{pmatrix}, \quad (\text{L3})$$

which does not fit the symmetries ruled by the cardinal equations. The action by the fully symmetric gauge yields

$$T' = \frac{1}{2} \begin{bmatrix} Ca + iSb + Cb + iSa & Ca + iSb + Cb + iSa \\ Ca + iSb - Cb - iSa & Ca + iSb - Cb - iSa \end{bmatrix}. \quad (\text{L4})$$

Applying the rules from Eqs. (K21) and (K22), the transfer tensor morphs into

$$\begin{cases} T'_{00} = (Ca + iSb + Cb + iSa)AO_p, \\ T'_{01} = (Ca + iSb + Cb + iSa)AO_i, \\ T'_{10} = (Ca + iSb - Cb - iSa)BO_p, \\ T'_{11} = (Ca + iSb - Cb - iSa)BO_i. \end{cases} \quad (\text{L5})$$

The A and B elements are, respectively, the idle and the interactive coefficients cast into the residual, whereas O_p and O_i are the idle and interactive coefficients for the output qubit. All these new coefficients, A , B , O_p , and O_i , are set to $1/\sqrt{2}$. We are interested in the sole T_{00} and T_{10} terms, the first representing the coefficient associated to the $\hat{I}$ basis element, the second representing the coefficient associated to the $\hat{Z}$ basis element. As no minus sign occurs between the interacting terms b and iS , the respective qubits are not neighbors. In contrast, a minus sign appears in front of the Cb and iSa terms, suggesting that such qubits neighbor with the $\hat{G}(A, B)$ one. For construction, the $\hat{G}(A, B)$ and the $\hat{G}(O_p, O_i)$ interact among them, and the overall graph state is therefore provided in Fig. 5(b). Within such scheme, the measurements over all the qubits are given by

$$\langle 0 | [\hat{I} + \hat{X}], \quad (\text{L6})$$

i.e., by a projection over the $|+\rangle$ state.

Nevertheless, it is possible to manipulate Eq. (L5) summing over $C + iS = e^{i\theta/2}$:

$$\begin{cases} T'_{00} = (ae^{i\theta/2} + be^{i\theta/2})AO_p, \\ T'_{01} = (ae^{i\theta/2} + be^{i\theta/2})AO_i, \\ T'_{10} = (ae^{-i\theta/2} - e^{-i\theta/2}b)BO_p, \\ T'_{11} = (ae^{-i\theta/2} - e^{-i\theta/2}b)BO_i. \end{cases} \quad (\text{L7})$$

It is possible to trace out an irrelevant global phase $e^{i\theta/2}$, making the transfer tensor equal to

$$\begin{cases} T'_{00} = (a + b)AO_p, \\ T'_{01} = (a + b)AO_i, \\ T'_{10} = (a - b)e^{-i\theta}BO_p, \\ T'_{11} = (a - b)e^{-i\theta}BO_i. \end{cases} \quad (\text{L8})$$

Applying the coefficient theorem, it is possible to split $e^{-i\theta}$ and B as two interactive coefficients, from a measurement

and from a PE, respectively, b interacting with B and therefore yielding the minus sign. The overall graph state is now described in Fig. 5(c), and the corresponding algebraic description, in terms of PEs, is provided by

$$[a + b\hat{X}_1\hat{Z}_2][A + B\hat{Z}_1\hat{X}_2\hat{Z}_3][O_p + O_i\hat{Z}_2\hat{X}_3]. \quad (\text{L9})$$

Recalling that $A = B = O_i = O_p = 1/\sqrt{2}$, the overall result is

$$\frac{1}{2}[a + b\hat{X}_1\hat{Z}_2][\hat{I} + \hat{Z}_1\hat{X}_2\hat{Z}_3][\hat{I} + \hat{Z}_2\hat{X}_3]. \quad (\text{L10})$$

The first qubit, pinned by $\hat{X}_1$, is measured in the $|+\rangle$ basis, whereas the second, provided by the $\hat{Z}_1\hat{X}_2\hat{Z}_3$ PE, is projected into the following state:

$$\langle 0 | [\hat{I} + e^{-i\theta}\hat{X}] \quad (\text{L11})$$

Equation (L10) is the stabilizer state to be measured, in order to achieve the transformation in Eq. (L1) (rotation around the x -axis) via the fully symmetric gauge.

APPENDIX M: EMPLOYING CLIFFORD OPERATIONS FOR GAUGE TRANSFORMATIONS

Beyond the transformations described by the cardinal equations, another set of gauge operations is provided by the elements from Clifford algebra. As stated in Appendix C, the elements belonging to the stabilizer group are closed with respect to the Clifford operations. Therefore, local Clifford operators can be employed even after the measurements, without attributing their action to the sole entanglement of the graph state. Rotations around the x -axis ($\hat{R}_x$) are a simple but still interesting example to showcase. In the first place, we embed the action of a $\hat{R}_x(\theta)$ gate in the PE formalism:

$$\hat{R}_x(\theta) |\psi\rangle = [\cos(\theta/2) + \sin(\theta/2)\hat{X}][a + b\hat{X}] |0\rangle. \quad (\text{M1})$$

Of course, such PEs do not fit the form of the residual in Eq. (J1). Nevertheless, it is possible to raise a $\hat{I} = \hat{H}^2$ operator in front of the kets, so that, employing the rules from Appendix C, the overall state now reads

$$\begin{aligned} & [\cos(\theta/2) + \sin(\theta/2)\hat{X}][a + b\hat{X}]\hat{H}^2 |0\rangle \\ &= \hat{H}[\cos(\theta/2) + \sin(\theta/2)\hat{Z}][a + b\hat{Z}]\frac{1}{\sqrt{2}}[\hat{I} + \hat{X}] |0\rangle, \end{aligned} \quad (\text{M2})$$

making use of $\hat{H} |0\rangle = 1/\sqrt{2}[\hat{Z} + \hat{X}] |0\rangle = 1/\sqrt{2}[\hat{I} + \hat{X}] |0\rangle$. The actual form now fits a possible residual.

Reversing the coefficient theorem, the second and third term are reshaped into

$$\begin{aligned} & [\cos(\theta/2) + \sin(\theta/2)\hat{Z}] \rightarrow \langle YZ(\theta) |_1 \frac{1}{\sqrt{2}}[\hat{I} + \hat{X}_1\hat{Z}] |0\rangle, \\ & [a + b\hat{Z}] \rightarrow \langle XY(0) |_2 [a + b\hat{X}_2\hat{Z}] |0\rangle \\ & \frac{1}{\sqrt{2}}[\hat{I} + \hat{X}] |0\rangle \rightarrow \frac{1}{\sqrt{2}}[\hat{I} + \hat{X}\hat{Z}_1\hat{Z}_2] |0\rangle, \end{aligned}$$

where $\langle YZ(\theta) |_1$ and $\langle XY(0) |_2$ belong to the measurement projections described in Appendix D, and the indices underline which space the $\hat{X}_i$ operators in Eq. (D1) act on. The graph state to be measured displays now as

$$\frac{1}{2}[\hat{I} + \hat{X}_1\hat{Z}][\hat{I} + \hat{X}\hat{Z}_1\hat{Z}_2][a + b\hat{X}_2\hat{Z}] |000\rangle. \quad (\text{M3})$$

After the $\langle YZ(\theta) |_1$, $\langle XY(0) |_2$ measurements, the $\hat{H}$ Hadamard operator needs to be applied on the output state.

APPENDIX N: FORMALIZATION OF THE EQUIVALENCE CLASS

The gauge freedom summarized by the cardinal equations makes straightforward to implement an equivalence relation between different graphs:

$$\begin{aligned} & G[V_1, E_1] \sim_U G[V_2, E_2] \Leftrightarrow \exists S : \mathfrak{T} \rightarrow \mathfrak{T} \mid T_i^1 = S_{ij}T_j^2 \\ & \vee T_i^2 = S_{ij}T_j^1, \\ & T_i^1T_j^1 = T_{i\circ j}^1T_{j\circ j}^1, \quad T_i^2T_j^2 = T_{i\circ j}^2T_{j\circ j}^2. \end{aligned} \quad (\text{N1})$$

Here G represents a simple graph, i.e., a graph with no self-loops nor multiple edges [62]. The sole distinction between these two G graphs is represented by the sets of vertices and edges $[V_i, E_i]$. The equivalence relation $\sim_U$ is given with respect to a certain unitary operator $\hat{U}$, meaning that the MBQC operations over $G[V_1, E_1]$ and $G[V_2, E_2]$ will yield the same outcome. Here $\mathfrak{T}$ stands for the space $[\mathbb{C}^4]^{\otimes r}$, r being the rank of T^1 and T^2 , the transfer tensors underlying the first and the second graphs. However, due to the potential singularity of the S tensor, it is mandatory to state that the S gauge operator could both map $T^1 \rightarrow T^2$ or $T^2 \rightarrow T^1$. The last constraint is pointed out by $T_i^1T_j^1 = T_{i\circ j}^1T_{j\circ j}^1$, as such condition endows T with the required symmetry to represent a graph in the stabilizer formalism. Once a unitary operator $\hat{U}$ can be implemented by a set of measurements over a graph $G[V, E]$, it follows that $G \in [U]$, i.e., G belongs to the class of graphs able to operate the $\hat{U}$ unitary transformation, thanks to the measurement techniques described in the MBQC theory.

APPENDIX O: MCALCULUS VERSUS FULLY SYMMETRIC GAUGE VERSUS LC-MBQC

Here the number of resource qubits involved by both the MCalculus and the fully symmetric gauge are compared. By invoking the rules from the MCalculus, the total number of qubits is left invariant for both optimized and unoptimized correction-measurement-entanglement (CME) pattern.

Another useful tool to settle a fair benchmark is provided by GRAPHIX [37], a software which leverages local Clifford operations in MBQC (LC-MBQC) and classical preprocessing of Pauli measurements in the measurement patterns.

In the MCalculus, every circuit, as can be visualized in Fig. 1, must be decomposed in terms of $\hat{J}$ and $\hat{CZ}$ gates, the representation of the $\hat{J}$ gate being given in Eq. (O7), and any $\hat{J}(\theta)$ gate can be decomposed as [63]

$$J(\theta) = X_2^{s_1} M_1^{-\theta} E_{1,2}. \quad (O1)$$

In this MCalculus notation (where we omit the hat over the operators), $E_{i,j}$ stands for the phase gate $\hat{CZ}_{ij}$, M^{theta} for the disruptive measurement $\dots$ as in $\dots$, and eventually the Pauli correction $\hat{X}$ depending on the outcome of the measurement. If the measurement is $|0\rangle$, then $s = 0$ and no byproduct is applied; otherwise, for $|1\rangle$, $s = 1$. A sequence of $J_n J_{n-1} \dots J_1$ yields $n + 1$ qubits and n edges:

$$X_{n+1}^{s_n} M_n^{\theta_n} E_{n,n+1} \dots X_2^{s_1} M_1^{\theta_1} E_{1,2}. \quad (O2)$$

Here, it is possible to shift all the E gates at the end of the queue (i.e., at the beginning of the computation), all the measurements in the mid and the corrections at the head of it. This sequence is called the CME pattern. Without getting into further details, the number of ancillary qubits and edges remains unaltered, which entails the fact that, given a sequence of gates acting on a single qubit in the circuit we want to translate into a graph state, we achieve n additional ancillary qubits and n edges on the corresponding graph.

Focusing instead on the fully symmetric gauge, we can deduce from Appendix L that the number of qubits N_{fully} is explained to scale as

$$N_{\text{fully}} = n_{\text{inputs}} + n_{\text{operations}} + 2n_{\text{inputs}} = 3n_{\text{inputs}} + n_{\text{operations}}. \quad (O3)$$

To calculate the total number of entangling operations required by the fully symmetric gauge, we perform numerical simulations. First, we generate the graph state corresponding to this choice of gauge, retrieving as output the number of edges e_k for the corresponding graph. Here, the k subscript refers to the number of input qubits n_k we choose (e.g., $n_1 = 1$, $n_2 = 2$, and so on). Since the process

does not involve the computation of the residual, as already stated in Sec. II C, when installing the graph state on a classical device no overhead occurs. The graphs are generated varying with the number of input qubits n and, for the QAOA, also along with the number of layers p . In the former case, we approximate the scaling law as a polynomial $a_i n^i$, and by plugging in some values n_k , we study the corresponding number of edges e_k :

$$a_i n_k^i = e_k. \quad (O4)$$

In this case, n_k^i forms a matrix with the i exponent and the subscript k as indices. By inverting the n_k^i matrix, it is straightforward to assess the scaling law for the number of entangling gates in the graph state:

$$a_i = (n_k^i)^{-1} e_k. \quad (O5)$$

As for the QAOA, we have to involve the scaling law for p as well. In this case, the polynomial assumes the form

$$a_{ij} n_k^i p_l^j = a_{ij} T_{ijkl} = e_{kl} \quad (O6)$$

Via some tensor calculus, it is thus possible to derive the complexity of the algorithm by computing a_{ij} . For the sake of completeness, we computed the edges e_k , e_{kl} of the graphs in the range $n \in \{2, 3, 4, 5, 6\}$ and $p \in \{1, 2, 3, 4, 5\}$. In all cases, we check that the maximum degree of the polynomial is bounded to 3. The numerical results are shown in Fig. 7.

Since GRAPHIX is an open-source PYTHON software, we adopted this empirical method to characterize the number of entangling gates and resource qubits involved by this method for the examined algorithms.

Example 1: Quantum Fourier transform

In order to provide an explicit example, a paramount and widespread algorithm is considered, consisting of the QFT. The QFT involves the applications of several controlled phase gates, whose number increases along with the number n of involved qubits. First, a decomposition of such controlled rotation gate is required. A valid model is provided in Fig. 6(b). Such circuit must therefore be converted into $\hat{J}(\phi)$ operators in order to perform the MCalculus, which is exposed in Fig. 6(c). The representation of the $\hat{J}(\phi)$ gate is given by [22]

$$\hat{J}(\phi) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & e^{i\phi} \\ 1 & -e^{i\phi} \end{pmatrix}. \quad (O7)$$

The conversion is straightforward, once we acknowledge the fact that

$$\hat{P}(\phi) = \hat{H} \hat{J}(\phi), \quad (O8)$$

where $\hat{P}$ is the phase gate. On the other hand, the fully symmetric gauge requires a decomposition in terms of Pauli

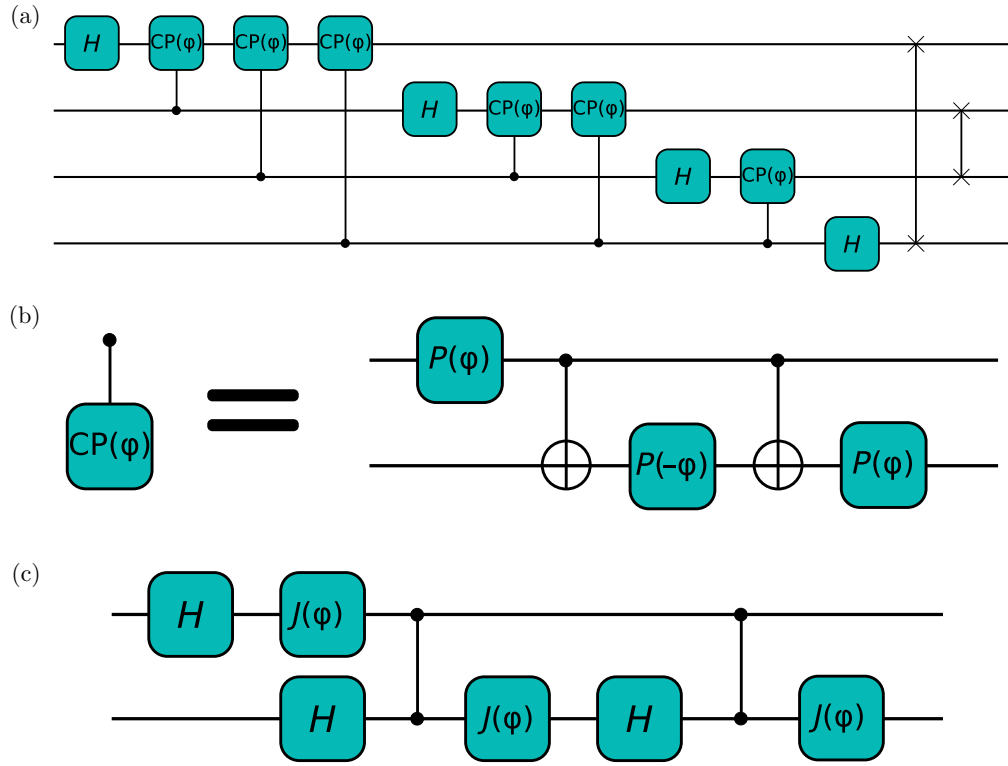


FIG. 6. (a) Scheme for the quantum Fourier transform (QFT) implemented with 4 qubits. (b) The controlled phase is decomposed in terms of Clifford gates and rotations. (c) Decomposition of the controlled-phase gate in terms of $\hat{J}(\phi)$ and $\hat{H}$ operators, $\hat{H}$ being $\hat{J}(0)$.

matrices. By applying the rules from the Clifford group, the controlled-phase gate turns out to be

$$\hat{C}P(\phi) = [\cos(\phi) + i \sin(\phi) \hat{Z}_1][\cos(\phi) + i \sin(-\phi) \hat{Z}_1 \hat{Z}_2] \times [\cos(\phi) + i \sin(\phi) \hat{Z}_2]. \quad (\text{O9})$$

Approaching the MCalculus, therefore, as $\hat{H} = \hat{J}(0)$, a qubit to be measured is involved per each Hadamard gate, whose number we refer to as n_H . Per each controlled-phase gate instead, as from Fig. 6(c), six measurements arise. Moreover, an additional qubit is required for any input qubit, whose total amount we refer to as n . Next, the final swaps need to be taken into account. The number of swap operations n_{swap} scales as $\lfloor n/2 \rfloor$ with respect with the total amount of qubits n , thus, in an asymptotic regime, as $n/2$. The swap gate can be decomposed into three CNOT gates:

$$\text{SWAP}_{ij} = \hat{C}X_{ij} \hat{C}X_{ji} \hat{C}X_{ij} \quad (\text{O10})$$

Due to the decomposition $\hat{C}X_{ij} = \hat{H}_j \hat{C}Z_{ij} \hat{H}_j$, the number of measurements in the MCalculus amounts to 6 per swap. In contrast, in the fully symmetric gauge, the CNOT gates can backpropagate at the head of the circuit, where $\hat{C}X_{ij} |0\rangle^{\otimes n} = |0\rangle^{\otimes n}$, transforming all of the $\hat{K}_i$, $\hat{K}_j$ stabilizers but without affecting the number of required qubits.

The overall scaling for the number of measurements $N_{\text{op}}^{\text{Mc}}$, varying with the number of input qubits n , is reported to be

$$N_{\text{op}}^{\text{Mc}} = n + n_H + n_{\text{CP}} \frac{n(n-1)}{2} + 6n_{\text{swap}} = 5n + 3n(n-1) \quad (\text{O11})$$

where $6n_{\text{swap}} = 6n/2 = 3n$, while $n_H = 1$. The $n(n-1)$ factor is linked to the number of controlled-phase operations, scaling from 1 to n for each qubit to be added. The overall summation, $\sum_{i=1}^n i$, returns indeed $n(n-1)/2$. Furthermore, the number of Hadamard gates n_H , as previously stated, equals the total number n of qubits. Switching to the fully symmetric gauge, the total number of operations amounts to n for the Hadamard gates, $3n_{\text{CP}}$ for the controlled-phase rotations, and $2n$ due to the peculiar structure of the gauge:

$$N_{\text{op}}^{\text{fully}} = 3n + \frac{3}{2}n(n-1). \quad (\text{O12})$$

Summing up the two scaling laws, we can eventually benchmark their complexity in terms of measurements and, therefore, involved qubits. As both the laws scale asymptotically with n^2 , it is straightforward to check that the MCalculus requires double the size of the graph state, $O(3n^2)$, compared with the fully symmetric gauge, $O(3n^2/2)$.

We now focus on the entanglement in the system, i.e., how many controlled-phase operations we need to perform. Referring to Fig. 6, the number of controlled rotations scales as $\sum_{i=1}^{n-1}$ with the number n of logic qubits. Referring to Fig. 6(c), we have two gates for the target qubit and four gates for the control one, plus two additional entangling gates. We set these parameters as $n_c = 4$, $n_t = 2$, and $n_e = 2$. The total account of operations then scales as

$$(n_c + n_t + n_e) \sum_{i=1}^{n-1} i = (n_c + n_t + n_e) \frac{n(n-1)}{2} = 4n(n-1). \quad (\text{O13})$$

Then, we need to add a single Hadamard gate for every logic qubit, and the final swaps. The contribution from the first is simply n , whereas the swaps, referring to (O10), involve two Hadamard gates and a controlled-phase one repeated three times, for a total of 2 additional entangling gates for mapping a single Hadamard in the CME pattern and three entangling gates inherited from the decomposition of the $\hat{C}\hat{X}$, the total number of $\hat{C}\hat{Z}$ amounting to $n_{\text{swap}} = 7$. The number of swaps is equal to $\lceil n/2 \rceil = O(n/2)$, and the computation of all the operations leads to

$$4n(n-1) + n + 7\lceil n/2 \rceil = O(4n^2). \quad (\text{O14})$$

As for the fully symmetric gauge, we numerical experiments return the following scaling law:

$$4n^2 - 4n + 2. \quad (\text{O15})$$

Therefore, no substantial changes occur switching from the MCalculus to the fully symmetric gauge, when focusing on the connectivity (i.e., the number of entangling gates) in the graph state.

To assess both the number of entangling gates and resource qubits involved by GRAPHIX, we exploit the numerical method detailed at the beginning of this appendix. All the values are reported in Table II.

Example 2: QAOA

To carry a new example, we now focus on the QAOA, a variational algorithm whose depth can scale at will with the number p of layers. When the QAOA is adapted to tackle the MaxCut problem, each layer is composed by a set of two operations, i.e., a two-qubit rotation $\hat{R}_{zz}$ and single-qubit rotation $\hat{R}_x$. In the MCalculus frame, the $\hat{R}_x$ rotations demand three qubits to be implemented:

$$\hat{R}_x\left(\frac{\theta}{2}\right) = \hat{J}(\theta)\hat{H} = X_3^s M_2^{-\theta} E_{2,3} X_2^s M_1^0 E_{1,2}. \quad (\text{O16})$$

On the other hand, each R_{zz} operation can be decomposed as follows:

$$\begin{aligned} \hat{R}_{z_i z_j}(\theta) &= [\cos(\theta) + i \sin(\theta) \hat{Z}_i \hat{Z}_j] \\ &= \hat{C}\hat{X}_{ij} [\cos(\theta) + i \sin(\theta) \hat{Z}_i] \hat{C}\hat{X}_{ij} \\ &= \hat{H} \hat{C}\hat{Z}_{ij} \hat{H} \hat{R}_{z_i}(\theta) \hat{H} \hat{C}\hat{Z}_{ij} \hat{H}_{ij}, \end{aligned} \quad (\text{O17})$$

from which 4 qubits are required by the CME decomposition, plus two additional input qubits. The fully symmetric gauge, instead, enables us to convert directly the $[\cos(\theta) + i \sin(\theta) \hat{Z}_i \hat{Z}_j]$ and $[\cos(\theta) + i \sin(\theta) \hat{X}_i]$ gates, already decomposed in terms of Pauli matrices. The number of $\hat{R}_{zz}$ operations, when approaching the MaxCut problem, depends on the total amount $\mathcal{E}$ of connections in the graph it has to be partitioned. We consider two limit cases, i.e., the cyclic graph and the complete graph. In the first case, $\mathcal{E} = n$, in the latter $\mathcal{E} = n(n-1)/2$. Hereafter, we can benchmark the complexity between the MCalculus and the fully symmetric gauge for the cyclic case:

$$\begin{cases} N_{\text{op}}^{\text{Mc}} = n + p(4n + 3n) = n(1 + 7p), \\ N_{\text{op}}^{\text{fully}} = 2n + 2np = 2n(1 + p). \end{cases} \quad (\text{O18})$$

Even for a number of layers $p = 1$, the MCalculus requires the double of qubits, whereas for $p \gg 1$ the ratio asymptotically scales to $7/2$. We now revert to the complete graph:

$$\begin{cases} N_{\text{op}}^{\text{Mc}} = n + 2pn(n-1) + 3pn, \\ N_{\text{op}}^{\text{fully}} = 2n + \frac{1}{2}pn(n-1) + np. \end{cases} \quad (\text{O19})$$

In such case, the leading term is n^2 , and for $n \gg 1$ the MCalculus requires 4 times the number of qubits with respect to the fully symmetric gauge.

As for the amount of controlled-phase gates in the circuit, we can state that, in the MCalculus, for each R_x rotation two controlled-phase gates are added, referring to Eq. (O16). This operation is repeated for every layer, and thus the number of entangling gates is equal to

$$n_e n p = 2np, \quad (\text{O20})$$

with n_e the number of entangling gates per each gate from the original, high-level formulation of the algorithm. As for the double Z rotation, the decomposition involves four $\hat{H}$ gates, a $\hat{R}_z(\phi) = \hat{H}\hat{J}(\phi)$ and two $\hat{C}\hat{Z}$, thus the total amount of entangling operations in the MCalculus is equal to 8. We need to add the number p of times the layer is repeated and the topology $\mathcal{E}$ of the MaxCut graph, leading to

$$8p\mathcal{E}. \quad (\text{O21})$$

Moreover, the Hadamard gates at the beginning at the circuit contribute by a n factor. Therefore, backing to

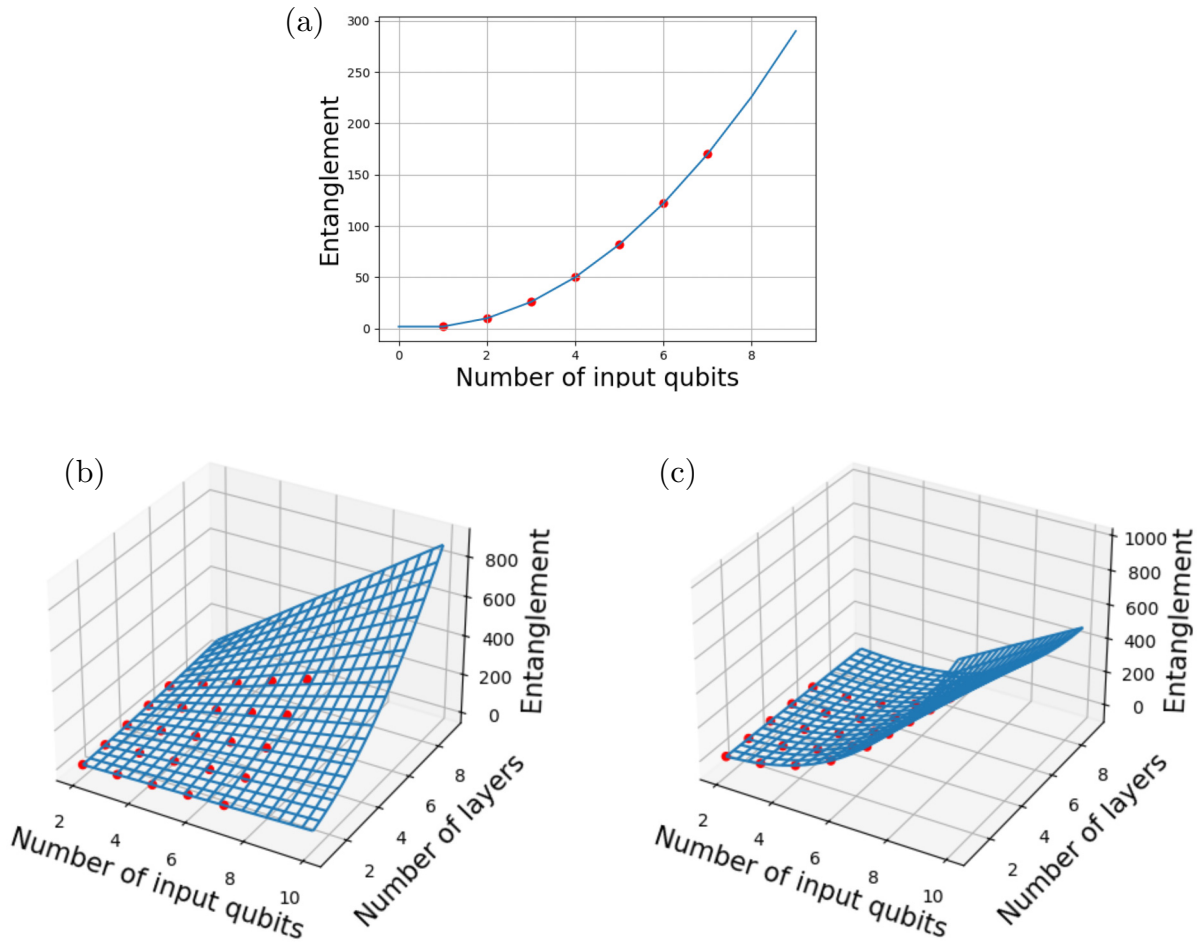


FIG. 7. An empirical characterization of the scaling laws for the entanglement in the fully symmetric gauge. The grid lines (in blue) represent the polynomials described in Eqs. (O4) and (O6), whereas the red dots represent the numerical values plugged into these algebraic equations to infer the polynomial coefficients. (a) Complexity laws for the entanglement in the QFT. (b) Complexity laws for the entanglement in the QAOA MaxCut for cyclic graphs. (c) Complexity laws for the entanglement in the QAOA MaxCut for complete graphs.

the topology of the MaxCut graph, $\mathcal{E} = n$ or $\mathcal{E} = n(n-1)/2$, depending the graph to be cyclic or complete. The corresponding scaling law for the entanglement, in the MCalculus, is for the cyclic topology equal to

$$E_{\text{op}}^{\text{Mc}} = n + 8pn + 2pn = n + 10pn, \quad (\text{O22})$$

whereas for complete graphs it reads

$$E_{\text{op}}^{\text{Mc}} = n + 8pn(n-1)/2 + 2pn. \quad (\text{O23})$$

To take into account the amount of entangling gates when adopting the fully symmetric gauge, we engage a numerical approach. In the first place, we assume the scaling law of the entanglement to be polynomial both in p and n , then we generate a set of values for p and n ranging, respectively, in $\{1, \dots, 5\}$ and $\{2, \dots, 6\}$. Next, we invert the tensor in Eq. (O6) in order to get the coefficients a_{ij} of the polynomial, since the amount of entangling gates e_{kl}

is retrieved from the numerical simulations. The results for the scaling laws of the QAOA can be visualized in Fig. 6(c) for the cyclic topologies and in Fig. 7(c) for the complete topologies. Globally, we observe that, for cyclic topologies, the asymptotic trend for the entanglement scales as $O(np^2)$, whereas for the complete topologies it is up to $O(n^3 + pn^2)$.

Again, the performances of GRAPHIX have been measured by a numerical approach. While GRAPHIX works slightly better for the cyclic topologies, we see that for complete topologies the number of entangling operations scales with n^2p^2 , where the MCalculus and the fully symmetric gauge scale with n^2p .

All the results are summed up in Table II.

-
- [1] R. Raussendorf and H. J. Briegel, A one-way quantum computer, *Phys. Rev. Lett.* **86**, 5188 (2001).

- [2] J. Wang, F. Sciarrino, A. Laing, and M. G. Thompson, Integrated photonic quantum technologies, *Nat. Photonics* **14**, 273 (2020).
- [3] G. Panichi, S. Corli, and E. Prati, Quantum physics-informed neural networks for multivariable partial differential equations, *Phys. Rev. Appl.* **25**, 014001 (2026).
- [4] S. Corli, L. Moro, D. Dragoni, M. Dispenza, and E. Prati, Quantum machine learning algorithms for anomaly detection: A review, *Future Gener. Comput. Syst.* **166**, 107632 (2025).
- [5] V. Dunjko, E. Kashefi, and A. Leverrier, Blind quantum computing with weak coherent pulses, *Phys. Rev. Lett.* **108**, 200502 (2012).
- [6] A. Kissinger and J. van de Wetering, Universal MBQC with generalised parity-phase interactions and Pauli measurements, *Quantum* **3**, 134 (2019).
- [7] J. Joo, C.-W. Lee, S. Kono, and J. Kim, Logical measurement-based quantum computation in circuit-QED, *Sci. Rep.* **9**, 16592 (2019).
- [8] J. R. Scott and K. C. Balram, Timing constraints imposed by classical digital control systems on photonic implementations of measurement-based quantum computing, *IEEE Trans. Quantum Eng.* **3**, 1 (2022).
- [9] M. Proietti, F. Cerocchi, and M. Dispenza, Native measurement-based quantum approximate optimization algorithm applied to the Max K-Cut problem, *Phys. Rev. A* **106**, 022437 (2022).
- [10] S. C. Benjamin, D. E. Browne, J. Fitzsimons, and J. J. Morton, Brokered graph-state quantum computation, *New J. Phys.* **8**, 141 (2006).
- [11] M. Zwerger, H. Briegel, and W. Dür, Hybrid architecture for encoded measurement-based quantum computation, *Sci. Rep.* **4**, 5364 (2014).
- [12] C. Kruszynska, A. Miyake, H. J. Briegel, and W. Dür, Entanglement purification protocols for all graph states, *Phys. Rev. A* **74**, 052316 (2006).
- [13] W. Dür, H. Aschauer, and H.-J. Briegel, Multiparticle entanglement purification for graph states, *Phys. Rev. Lett.* **91**, 107903 (2003).
- [14] H. Aschauer, W. Dür, and H.-J. Briegel, Multiparticle entanglement purification for two-colorable graph states, *Phys. Rev. A* **71**, 012319 (2005).
- [15] S. Glancy, E. Knill, and H. M. Vasconcelos, Entanglement purification of any stabilizer state, *Phys. Rev. A* **74**, 032319 (2006).
- [16] A. Bera, F. A. Wudarski, G. Sarbicki, and D. Chruściński, Class of Bell-diagonal entanglement witnesses in $C^4 \otimes C^4$: Optimization and the spanning property, *Phys. Rev. A* **105**, 052401 (2022).
- [17] N. C. Menicucci, Fault-tolerant measurement-based quantum computing with continuous-variable cluster states, *Phys. Rev. Lett.* **112**, 120504 (2014).
- [18] M. V. Larsen, X. Guo, C. R. Breum, J. S. Neergaard-Nielsen, and U. L. Andersen, Deterministic generation of a two-dimensional cluster state, *Science* **366**, 369 (2019).
- [19] R. Raussendorf, J. Harrington, and K. Goyal, Topological fault-tolerance in cluster state quantum computation, *New J. Phys.* **9**, 199 (2007).
- [20] A. Mantri, T. F. Demarie, and J. F. Fitzsimons, Universality of quantum computation with cluster states and (X, Y) -plane measurements, *Sci. Rep.* **7**, 42861 (2017).
- [21] S. Corli, D. Dragoni, M. Proietti, M. Dispenza, C. Cavazzoni, and E. Prati, Benchmarking the emulation of measurement-based quantum computing through the max k -cut algorithm, *Quantum Inf. Process.* **24**, 396 (2025).
- [22] V. Danos, E. Kashefi, and P. Panangaden, The measurement calculus, *J. ACM* **54**, 8 (2007).
- [23] E. Pius, in *High Performance Computing* (Edinburgh, 2010), <https://static.epcc.ed.ac.uk/dissertations/hpc-msc/2009-2010/Einar%20Pius.pdf>.
- [24] R. Duncan and S. Perdrix, in *International Colloquium on Automata, Languages, and Programming* (Springer, 2010), p. 285, https://dl.wqtxts1x7le7.cloudfront.net/30783752/gflow-libre.pdf?1391869645=&response-content-disposition=inline%3B+filename%3DRewriting_measurement_based_quantum_comp.pdf&Expires=1764940561&Signature=ODUW8g3DmQy4FVr3aF5HuUcUVZiGQQTQfmuwLqZZPIfA0ZjLHVJGLe46z5zqmU3k0HeBeFLxme-lczhybls3PII-NYbdroP13R50IdRIIJorvAgjHjPYBTYCNfKGygGIXUXxVjJX66geIyfcIQ3t9Owh9N92dNJcpsXIWZkhe2HHio-xhWdhrwn4tWU3X9MJzBJZoRuTnKay1~KtOBmrg56GB4vfAxFZXY7oLEcsKxm4BzoT66x5KD269viubhB46YJIZg2EPQrDzmx1wUZ7NTQiONEZI~zl8SwPbR7aqUMQBMt2R-ZDAYtwluH~laCmu6z59wJWBfgv3JQBla7lw__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA.
- [25] W. Simmons, Relating measurement patterns to circuits via Pauli flow, *arXiv:2109.05654*.
- [26] J. Miyazaki, M. Hajdušek, and M. Murao, Analysis of the trade-off between spatial and temporal resources for measurement-based quantum computation, *Phys. Rev. A* **91**, 052302 (2015).
- [27] M. Backens, H. Miller-Bakewell, G. de Felice, L. Lobski, and J. van de Wetering, There and back again: A circuit extraction tale, *Quantum* **5**, 421 (2021).
- [28] B. Kraus, Local unitary equivalence of multipartite pure states, *Phys. Rev. Lett.* **104**, 020504 (2010).
- [29] N. Claudet and S. Perdrix, *STACS: 42nd International Symposium on Theoretical Aspects of Computer Science* (2025).
- [30] M. K. Vijayan, A. Paler, J. Gavriel, C. R. Myers, P. P. Rohde, and S. J. Devitt, Compilation of algorithm-specific graph states for quantum circuits, *Quantum Sci. Technol.* **9**, 025005 (2024).
- [31] R. R. Ferguson, L. Dellantonio, A. A. Balushi, K. Jansen, W. Dür, and C. A. Muschik, Measurement-based variational quantum eigensolver, *Phys. Rev. Lett.* **126**, 220501 (2021).
- [32] F. K. Marquers and N. T. Zinner, Applications and resource reductions in measurement-based variational quantum eigensolvers, *Quantum Sci. Technol.* **8**, 045001 (2023).
- [33] T. N. Kaldenbach and M. Heller, Mapping quantum circuits to shallow-depth measurement patterns based on graph states, *Quantum Sci. Technol.* **10**, 015010 (2025).
- [34] H. Poulsen Nautrup and H. J. Briegel, Measurement-based quantum computation from clifford quantum cellular automata, *Phys. Rev. A* **110**, 062617 (2024).
- [35] J. Ruh and S. Devitt, Quantum circuit optimisation and mbqc scheduling with a pauli tracking library, *IEEE Trans. Quantum Eng.* (2025).
- [36] A. Paler, S. J. Devitt, K. Nemoto, and I. Polian, Software Pauli tracking for quantum computation, *arXiv:1401.5872*.

- [37] S. Sunami and M. Fukushima, Graphix: Optimizing and simulating measurement-based quantum computation on local-Clifford decorated graph, [arXiv:2212.11975](#).
- [38] Y. Li, A. Pawar, Z. Mo, Y. Zhang, J. Yang, and X. Tang, in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 4* (ACM Digital Library, New York, 2024), p. 111.
- [39] S. Liu, N. Benchasattabuse, D. Q. Morgan, M. Hajdušek, S. J. Devitt, and R. Van Meter, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Vol. 1 (IEEE, 2023), p. 870, <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10313779>.
- [40] J. van de Wetering, ZX-calculus for the working quantum computer scientist, [arXiv:2012.13966](#).
- [41] D. Browne and H. Briegel, One-way Quantum Computation, Quantum information: From foundations to quantum technology applications, [arXiv:quant-ph/0603226](#).
- [42] G. Wong, R. Raussendorf, and B. Czech, The gauge theory of measurement-based quantum computation, *Quantum* **8**, 1397 (2024).
- [43] S. Corli and E. Prati, in *Journal of Physics: Conference Series*, Vol. 3017 (IOP Publishing, 2025), p. 012043.
- [44] S. Corli and E. Prati, in *Quantum 2.0* (Optica Publishing Group, San Francisco, 2025), p. QW4B–5.
- [45] S. Corli and E. Prati, in *2022 IEEE International Conference on Rebooting Computing (ICRC)* (IEEE, 2022), p. 1, <https://arxiv.org/pdf/2212.12102.pdf>.
- [46] N. Khaneja and S. J. Glaser, Cartan decomposition of $SU(2^n)$ and control of spin systems, *Chem. Phys.* **267**, 11 (2001).
- [47] V. Saravanan and S. M. Saeed, in *2021 22nd International Symposium on Quality Electronic Design (ISQED)* (IEEE, 2021), p. 73, <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9424311>.
- [48] A. Rodriguez, *KAK decomposition in quantum computing* (Lamar University-Beaumont, 2012), https://www.proquest.com/openview/4a4aef1b8c2c4abc7a91f4d525ea3920/1?casa_token=PqbeLo7V-WsAAAAA:jOjh9XINQISevf5u9V5Lxy0hPDJ2O57M6isuMyCpHNRmps4vdqE1ghACtLSBeAaNz83gsVFeqD3I&cbl=18750&pq-origsite=gscholar&parentSessionId=Z12ToMLMJBPXbZCcwYfFbg4zmo9tch7t9rdeC%2B9KLIk%3D.
- [49] L. Lao, A. Korotkov, Z. Jiang, W. Mruczkiewicz, T. E. O'Brien, and D. E. Browne, Software mitigation of coherent two-qubit gate errors, *Quantum Sci. Technol.* **7**, 025021 (2022).
- [50] S. S. Bullock and G. K. Brennen, Canonical decompositions of n-qubit quantum computations and concurrence, *J. Math. Phys.* **45**, 2447 (2004).
- [51] T. N. Kaldenbach, I. D. Smith, H. P. Nautrup, M. Heller, and H. J. Briegel, Efficient preparation of resource states for Hamiltonian simulation and universal quantum computation, [arXiv:2509.05404](#).
- [52] J. C. Adcock, C. Vigliar, R. Santagati, J. W. Silverstone, and M. G. Thompson, Programmable four-photon graph states on a silicon chip, *Nat. Commun.* **10**, 3528 (2019).
- [53] D. E. Browne, E. Kashefi, M. Mhalla, and S. Perdrix, Generalized flow and determinism in measurement-based quantum computation, *New J. Phys.* **9**, 250 (2007).
- [54] D. Markham and E. Kashefi, Entanglement, flow and classical simulatability in measurement based quantum computation, *Horizons of the Mind. A Tribute to Prakash Panangaden: Essays Dedicated to Prakash Panangaden on the Occasion of His 60th Birthday*, [arXiv:1311.3610](#).
- [55] N. M. Linke, D. Maslov, M. Roetteler, S. Debnath, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe, Experimental comparison of two quantum computing architectures, *Proc. Natl. Acad. Sci.* **114**, 3305 (2017).
- [56] J. Dehaene and B. De Moor, Clifford group, stabilizer states, and linear and quadratic operations over GF (2), *Phys. Rev. A* **68**, 042318 (2003).
- [57] M. Van den Nest, J. Dehaene, and B. De Moor, Local unitary versus local Clifford equivalence of stabilizer states, *Phys. Rev. A* **71**, 062323 (2005).
- [58] S. Anders and H. J. Briegel, Fast simulation of stabilizer circuits using a graph-state representation, *Phys. Rev. A* **73**, 022334 (2006).
- [59] N. Rengaswamy, R. Calderbank, S. Kadhe, and H. D. Pfister, Logical Clifford synthesis for stabilizer codes, *IEEE Trans. Quantum Eng.* **1**, 1 (2020).
- [60] P. Kok, in *Theoretical Foundations of Quantum Information Processing and Communication: Selected Topics* (Springer, Heidelberg, 2009), p. 187.
- [61] C. G. Brell, Generalized cluster states based on finite groups, *New J. Phys.* **17**, 023029 (2015).
- [62] D. B. West, *Introduction to Graph Theory* (Prentice Hall, Upper Saddle River, NJ, 2001), Vol. 2, <https://www.academia.edu/download/6479067/igtpref.ps>.
- [63] S. Corli, *Benchmarking quantum machine learning for quantum cybersecurity*, Ph.D. thesis (2025).