



Deposited via The University of Sheffield.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/244478/>

Version: Published Version

Article:

Rytir, P., Burke, P.C., Aravanis, C. et al. (2026) Topological quantum compilation using mixed-integer programming. IEEE Transactions on Quantum Engineering, 7. 3104212. ISSN: 2689-1808

<https://doi.org/10.1109/tqe.2026.3710775>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

Received 25 January 2026; revised 30 June 2026; accepted 1 July 2026; date of publication 7 July 2026;
date of current version 4 August 2026.

Digital Object Identifier 10.1109/TQE.2026.3710775

Topological Quantum Compilation Using Mixed-Integer Programming

PAVEL RYTÍŘ¹ , PHILLIP C. BURKE^{2,3} , CHRISTOS ARAVANIS^{4,1},
JIRÍ VALA^{5,6,7} , AND JAKUB MAREČEK¹ 

¹Department of Computer Science, Czech Technical University in Prague, 12135 Prague, Czech Republic

²School of Physics, University College Dublin, D04 P7W1 Dublin, Ireland

³Centre for Quantum Engineering, Science, and Technology, University College Dublin, D04 V1W8 Dublin, Ireland

⁴Sheffield International College, University of Sheffield S1 4DE, U.K.

⁵Department of Physics, Maynooth University, W23 X021 Maynooth, Kildare, Ireland

⁶School of Theoretical Physics, Dublin Institute for Advanced Studies, D04 C932 Dublin, Ireland

⁷Tyndall National Institute, T12 R5CP Cork, Ireland

Corresponding author: Jakub Mareček (e-mail: jakub.marecek@fel.cvut.cz)

This work was supported in part by Czech Science Foundation under Grant 23-0947S and in part by Taighde Éireann - Research Ireland under Grant 21/RP-2TF/10019.

ABSTRACT Quantum computation offers the potential to solve certain problems faster than classical computing, but its practical realization is hindered by significant challenges associated with error propagation and decoherence. Topological quantum computing addresses these issues by encoding qubits in particular 2-D topological condensed matter systems, effectively described by topological quantum field theories, where quantum gates are implemented through sequences of braids of quasi-particles with exotic fractional statistics. The explicit construction of these sequences is a nontrivial task. In this work, we introduce a general optimization framework for a broad class of quantum compilation problems, including the compilation of quantum gates in topological quantum computing. The framework is based on Mixed-Integer Quadratically-Constrained Quadratic Programming, which employs convex relaxations to tackle the underlying nonconvex optimization problem. We demonstrate the effectiveness of this framework by compiling a quantum gate from the local equivalence class of the controlled-NOT operation for a nonsemisimple topological field theory for the first time. Our results provide a constructive pathway from abstract universality proofs to explicit gate implementations, as a practical tool in topological quantum compilation.

INDEX TERMS Compilers, mixed integer linear programming, quantum circuit, quantum computing, quadratic programming, optimizing compilers, topological quantum compilation.

I. INTRODUCTION

Quantum computation [1] can solve an important class of computational problems with a considerable speed-up compared to its classical counterpart. Nevertheless, it faces important challenges due to quantum errors in the computation process. In order for quantum computation to achieve its full potential, it has to go beyond the intrinsic limitations of noisy intermediate-scale quantum devices [2]. This requires a complex fault-tolerant quantum computing architecture [3], which is based on quantum error correction with strict error-correction thresholds [4], [5], [6].

Topological quantum computation [7], [8], [9], [10] promises to bypass these requirements by storing and processing quantum information in topological quantum states. These states are effectively described by topological quantum field theory, from which they inherit robustness against

errors, such as those arising from undesired interactions with stray particles or physical fields. When realized, for example, in a 2-D many-body quantum system, these states correspond to ground states with a finite degree of degeneracy, which depends on topological properties of the underlying 2-D manifold, such as its genus, i.e., the number of defects, e.g., punctures or the number of quasi-particle excitations. The physical realization of topological quantum computation is an area of intense research activity. Recent examples relevant to our work include, for instance, in [11], utilizing superconducting circuits, and [12], utilizing ion traps. Although these works are strictly speaking simulations, they offer experimental platforms for demonstrating quasiparticle braiding and fractional statistics in non-Abelian systems.

Point-like excitations, specifically in 2-D topological systems, are known as anyons. These excitations are

characterized by fractional quantum statistics, which are linked to irreducible representations of the braid group [13]. Braiding of these anyons results in a transformation of the quantum state in the anyonic system, which can be either Abelian or non-Abelian. In the non-Abelian case, this transformation corresponds to a unitary rotation of a state vector within the manifold of degenerate ground states of the topological many-body system. The braiding of anyons thus implements quantum computing operations on n quantum bits or qubits, encoded in the ground-state manifold with Hilbert space dimension $d \geq 2^n$. The gate alphabet, native to a topological system, is hence given by a set of elementary braids between non-Abelian anyons. An important feature of anyon theory from the point of view of quantum computation is universality, that is, whether braiding supports universal quantum computing operations from the unitary group $U(2^n)$ over the Hilbert space of n quantum bits.

The compilation problem is a key element in the successful implementation of quantum computation on specific quantum hardware. Its objective is to convert a quantum algorithm or protocol into a form that can then be executed on a quantum computing device. Quantum hardware is typically characterized by various constraints, such as qubit connectivity and native gate alphabet. The goal of topological quantum compiling is to design a specific sequence of the elementary braids in order to implement a desired quantum computing gate, which unitarily transforms a qubit state encoded into a multiply degenerate ground state of a given topological system.

The seminal works on topological quantum compilation [14], [15], [16] primarily use a three-anyon encoding of a logical qubit to realize both single-qubit and two-qubit operations via braiding or “weaving” of Fibonacci anyons. In more recent work, Carnahan et al. [17] developed a systematic approach to the generation of two-qubit anyon braids based on an iterative procedure, which was proposed earlier by Reichardt [18]. Kliuchnikov et al. [19] developed an asymptotically optimal probabilistic polynomial algorithm to generate single-qubit gates exactly and also to produce depth-optimal approximations of two-qubit gates in the Fibonacci anyon model. Relevant to the topological quantum compilation problem is also the work on topological hashing by Burrello et al. [20] and a recent application of reinforcement learning for quantum compilation [21], [22].

The implementation of gates in topological quantum computing typically requires elementary braiding operations acting on a Hilbert space of a larger dimension than that of two qubits. This may lead to the loss of quantum information out of the computational subspace into the rest of the Hilbert space; this process is known as *leakage*. The problem of leakage in relation to topological quantum compilation, particularly in the context of the implementation of two-qubit gates, was recognized early on by Bonesteel et al. [14]. This problem was studied later by Xu and Wan [23], and also by Ainsworth and Slingerland [24], who focused on the leakage

problem in connection with topological qubit design and showed that the requirement of leakage-free design is too restrictive and can only be realized in nonuniversal Ising-like anyon models. A large family of leakage-free Fibonacci braiding gates were studied by Cui et al. [25], who proved that they were all nonentangling. Moreover, brute-force numerical searches failed to identify any leakage-free implementations of entangling operations using braids with a word length of up to seven.

Recently, Iulianelli et al. [26] proposed a new framework for topological quantum computation based on nonsemisimple extensions of $(2 + 1)$ -dimensional topological quantum field theories. These enhanced models go beyond conventional Ising anyons, which are associated with the $\nu = \frac{5}{2}$ fractional quantum Hall state [27], but which are not universal for quantum computation via braiding alone. In the nonsemisimple theory, the Ising framework is extended by a single new anyon, which enables universal quantum computation solely through braiding, thus lifting the Ising anyon system to the universal topological quantum computation.

In this work, we provide a Mixed Integer Quadratically Constrained Quadratic Programming (MIQCQP) formulation for the gate compilation problem. As a case study, we demonstrate our approach on a system of anyons that emerge in a nonsemisimple version of the Ising topological quantum field theory [26]. We specifically aim to compile two-qubit entangling gates, such as the controlled-NOT (CNOT) gate and its local equivalence class, as well as other perfect entanglers. We employ the geometric theory of two-qubit operations, developed by Zhang et al. [28] in the context of quantum control. The geometric theory stems from the Cartan decomposition of the Lie group $SU(4)$ and the Makhlin local invariants [29], which provides a representation of two-qubit operations in terms of their local equivalence classes. In our context, a local equivalence class is defined as the set of two-qubit gates that are equivalent up to single-qubit transformations, and thus correspond to the co-set of $SU(4)$ by the subgroup $SU(2) \otimes SU(2)$. We will use the gate alphabet proposed in the paper [26].

The recent work by Burke et al. [30] significantly extended the calculations by Cui et al. [25] to larger braiding combinations and also employed the theory of local invariants and equivalence classes of two-qubit operations. The local invariants were previously used by DiVincenzo et al. [31] to find a local equivalent of the CNOT gate in a system with the Heisenberg Hamiltonian in the context of encoded universality. Müller et al. [32] have combined optimal quantum control with the geometric theory of two-qubit operations to derive an optimization algorithm that determines the best two-qubit entangling gate in the system of trapped polar molecules and neutral atoms. This work was later followed by a generalization of the optimal control objective to include an arbitrary perfect entangler [33], [34]. In all these cases, the use of geometric theory, and the Makhlin local invariants in particular, led to a significant enlargement of the optimization target, its dimension, and size, which

considerably improved both the accuracy and convergence of the optimization process.

The rest of this article is organized as follows. In Section II, we present the mixed-integer programming formulation of quantum compilation. Section III briefly reviews the main features of the system of Ising anyons in a non-semisimple topological quantum field theory. Section IV presents the experimental results of topological quantum compilation in this system. Finally, Section V concludes this article.

II. MIXED-INTEGER PROGRAMMING FORMULATION

The goal of this section is to formulate the quantum compilation problem in the MIQCQP framework [35], [36], [37]. This optimization technique is particularly suitable for optimization problems, in which some of the variables are discrete. The main advantage of this method, from our perspective, is that it provides global optimality guarantees. This, in turn, prevents the braid sequences resulting from the compilation of entangling quantum operations from being unnecessarily long. Another advantage of MIQCQP is that it has been continuously developed and extensively utilized, for example, in optimizing logistics problems. Hence, there are many mature MIQCQP solvers with global optimality guarantees available. This also makes the method well understood, including its computational complexity. MIQCQP is, in general, as hard or harder than any problem in the class of nondeterministic polynomial time (NP-hard), but the computational complexity can be more favorable in specific instances. Moreover, its widespread usage carries over to the quantum compilation problem when formulated in the MIQCQP framework. In the following subsections, we provide a brief outline of the quantum compilation problem and then give a detailed description of the MIQCQP formulation in the quantum compilation context.

A. QUANTUM COMPILATION PROBLEM

The exact (i.e., not locally equivalent) quantum compilation problem aims to find a sequence of gates from a specified alphabet of operations or gates that equals the desired target operation T . The quantum compilation problem can be formally defined as follows: Let $\mathcal{A} = \{\rho_1, \dots, \rho_m\} \in \mathbb{C}^{n \times n}$ be the matrices representing the gate alphabet, and $T \in \mathbb{C}^{n_C \times n_C}$ be a target gate, with $n_C \leq n$; $n_C, n \in \mathbb{N}$. Here, n_C is the dimension of the computational subspace, V_C . Let $d \in \mathbb{N}$ and $I: \{1, \dots, d\} \mapsto \{1, \dots, m\}$ be a function encoding a sequence of indices of gates from alphabet $\mathcal{A}$. We define the unitary matrix P as the resultant operation corresponding to a sequence I as

$$P = \prod_{i=1}^d \rho_{I(i)}. \quad (1)$$

A unitary operation P is said to be leakage-free if it maps the computational subspace V_C onto itself. Thus, a leakage-free

operation must have the block-diagonal structure

$$P = X \oplus U \quad (2)$$

where U is a unitary matrix acting on the computational subspace V_C with dimension n_C , and X is a unitary matrix acting on the non-computational subspace V_{NC} of dimension n_{NC} . The total dimension of the Hilbert space is thus $n = n_C + n_{NC}$.

From a practical point of view, it is useful to relax the notion of exact quantum compiling to its approximate form, in which one or several objectives are achieved within a desired accuracy. This may include the approximation of the target gate T , or approximate unitarity (or approximate leakage-free realization) in the computational subspace V_C within an accuracy given by some suitable distance metric. We will specify these in a concrete example below.

B. PROGRAM VARIABLES

We shall assume that the circuit has a fixed depth d . If we do not know the exact depth in advance (which is the case for our compilation problem), we can include an identity matrix in the alphabet, permitting us to search for a circuit of maximum depth d . With d the fixed depth of the circuit, the compiled circuit is constructed in d steps. For $t = 0, \dots, d$; we define the complex variables $[Y_t]_{i,j} \in \mathbb{C}$ for $i, j = 1, \dots, n$. These variables constitute the matrix Y_t , which encodes the circuit obtained as a product of t gates from the alphabet, i.e.,

$$Y_t = \prod_{i=1}^t \rho_{I(i)} \quad (3)$$

where I is any sequence of gates determined by the constraint (5).

At each step $t = 1, \dots, d$, we have to decide which gate from the alphabet of size m should be applied. This decision is encoded as a binary decision variable $x_{t,i} \in \{0, 1\}$, for $t = 1, \dots, d$ and $i \in 1, \dots, m$. Here, $x_{t,i} = 1$ if, and only if, gate ρ_i is applied at step t , in which case, $x_{t,j} = 0 \ \forall j \neq i$.

C. PROGRAM CONSTRAINTS

First, we must define the initial state constraints. These constraints say that the empty product of gates from the alphabet equals the $n \times n$ identity matrix

$$Y_0 = \mathbb{1}_n. \quad (4)$$

In each step $t = 1, \dots, d$, we multiply the product Y_{t-1} , constructed in the previous step $t - 1$, by the chosen gate from the alphabet. The relation between Y_t and Y_{t-1} is encoded in the following matrix constraint:

$$Y_t = \left(\sum_{i=1}^m x_{t,i} \rho_i \right) Y_{t-1} \quad (5)$$

where $x_{t,i}$ are the binary decision variables, selecting which gate from the alphabet is applied at step t . We reiterate that exactly one gate can be applied in each step. This is enforced

by the following constraint:

$$\sum_{i=1}^m x_{t,i} = 1, \quad \text{for every step } t = 1, \dots, d. \quad (6)$$

Finally, we introduce constraints that restrict leakage to the noncomputational space of the final product Y_d . We wish to enforce that $Y_d = X \oplus U$, where U is the submatrix that corresponds to the computational space. By setting the elements of Y_d outside the blocks X and U to zero. Then, the matrix U has to be unitary as Y_d is unitary, and thus there is no leakage to the noncomputational space. We recall that an $m \times m$ matrix U is defined to be unitary if $U^\dagger U = U U^\dagger = \mathbb{1}$.

D. PROGRAM OBJECTIVE FUNCTION

We now need to define the objective function for the problem. Due to numerical errors and inaccuracies, it is unlikely that we could find a product of the alphabet gates that is exactly equal to the target gate. Therefore, we define the distance J between the target gate and the computational part of the computed product Y_d . The distance J is defined as the square of the Frobenius norm of the difference between T and U

$$J(T, U) = \|T - U\|_F^2 = \sum_{k,\ell=1}^n |T_{k,\ell} - U_{k,\ell}|^2. \quad (7)$$

Our goal will be to minimize this distance, i.e., this is our objective function.

E. MIQCQP PROBLEM FORMULATION

The mixed integer quadratic program for the gate compilation problem is defined as follows. Let Y_d^C denote the computational part of the matrix Y_d at the final step d .

Our objective is to minimize

$$J(T, Y_d^C) = \sum_{k,\ell=n_{\text{NC}}+1}^n |T_{k,\ell} - [Y_d]_{k,\ell}|^2. \quad (8)$$

The optimization procedure is subject to the following constraints:

$$\text{i) } [Y_t]_{k,\ell} = \sum_{p=1}^n \left(\sum_{i=1}^m x_{t,i} [\rho_i]_{k,p} [Y_{t-1}]_{p,\ell} \right), \quad (9)$$

for $k, \ell = 1, \dots, n$, and $t = 1, \dots, d$

$$\text{ii) } \sum_{i=1}^m x_{t,i} = 1, \quad \text{for each } t = 1, \dots, d \quad (10)$$

$$\text{iii) } x_{t,i} \in \{0, 1\}, \quad \text{for } t = 1, \dots, d, \text{ and } i = 1, \dots, m \quad (11)$$

$$\text{iv) } [Y_0]_{k,\ell} = \delta_{k,\ell}, \quad \text{for } k, \ell = n_{\text{NC}} + 1, \dots, n \quad (12)$$

$$\text{v) } [Y_d]_{k,\ell} = 0, \quad \text{for } k = 1, \dots, n_{\text{NC}}; \ell = n_{\text{NC}} + 1, \dots, n \quad (13)$$

$$\text{vi) } [Y_d]_{k,\ell} = 0, \quad \text{for } k = n_{\text{NC}} + 1, \dots, n; \ell = 1, \dots, n_{\text{NC}} \quad (14)$$

$$\text{vii) } [Y_t]_{k,\ell} \in \mathbb{C}, \quad \text{for } t = 0, \dots, d, \text{ and } k, \ell = 1, \dots, n. \quad (15)$$

The constraints (i) correspond elementwise to (5), while the constraints (ii) correspond to (6). The constraints (iii) enforce integrality on the gate selection variables, and (iv) set the initial empty product to be the $n \times n$ identity matrix. The constraints (v) and (vi) enforce the final product Y_d to have the form $X \oplus Y_d^C$, where X acts on the noncomputational space and Y_d^C on the computational space. These constraints also ensure that Y_d^C to be unitary, since Y_d is unitary. Finally, the constraints (vii) specify that the product variables are complex numbers.

F. MIQCQP PROBLEM RELAXATION AND LINEARIZATION

The performance of solvers on MIQCQP programs is usually more time-consuming than on mixed-integer linear programs (MILP). This is due to three factors: the root relaxation of an MIQCQP is being solved with an interior-point method that cannot exploit as much sparsity as in the MILP case, and may need a cross-over routine. Relaxations solved subsequently in the branch-and-bound tree are also more difficult to solve, as the dual simplex commonly used in MILP does not generalize easily. Finally, there are stronger cuts applicable to MILP than MIQCQP, which can increase the number of nodes visited in the branch-and-bound process [38]. In proof complexity [39], [40], one could provide lower bounds for the final aspect.

The constraints in (5) and the objective function (7) are nonlinear. This can negatively affect the performance of the solver. The McCormick linearization procedure [41] replaces a bilinear constraint $z = ab$, where a is a binary variable and b is a continuous variable, by the following set of linear constraints. Let $b_{\min}$ and $b_{\max}$ be the minimal and maximal allowed values of b , respectively

$$\begin{aligned} z &\geq b_{\min} a \\ z &\geq b + b_{\max} a + b_{\min} \\ z &\leq b_{\max} a \\ z &\leq b + b_{\min} a + b_{\max}. \end{aligned} \quad (16)$$

This linearization is exact; replacing the bilinear constraint $z = ab$ with the four linear constraints (16) yields the exact same set of solutions.

Using this McCormick linearization, all constraints in (9) can be replaced by a set of linear constraints. The right-hand side of (9) is the sum of the bilinear expressions

$$x_{t,i} [\rho_i]_{k,p} [Y_{t-1}]_{p,\ell} \quad (17)$$

where $[\rho_i]_{k,p}$ is a constant. We apply McCormick linearization to transform the bilinear constraint in (9) into a set of linear constraints without worsening the solution quality, as follows. For each $i = 1, \dots, m$; $k = 1, \dots, n$; $\ell = 1, \dots, n$; $p = 1, \dots, n$; and $t = 1, \dots, d$, we introduce a new variable, $z_{t,k,\ell}^{p,i}$, which is constrained to take the same value as the

expression in (17). For each $z_{t,k,l}^{p,i}$, we add the following four constraints:

$$\begin{aligned} z_{t,k,l}^{p,i} &\geq -|[\rho_i]_{k,p}|x_{t,i} \\ z_{t,k,l}^{p,i} &\geq [\rho_i]_{k,p}[Y_{t-1}]_{p,l} + |[\rho_i]_{k,p}|x_{t,i} - |[\rho_i]_{k,p}| \\ z_{t,k,l}^{p,i} &\leq |[\rho_i]_{k,p}|x_{t,i} \\ z_{t,k,l}^{p,i} &\leq [\rho_i]_{k,p}[Y_{t-1}]_{p,l} - |[\rho_i]_{k,p}|x_{t,i} + |[\rho_i]_{k,p}|. \end{aligned} \quad (18)$$

Finally, the constraint of (9) is replaced by the following linear constraint:

$$[Y_t]_{k,\ell} = \sum_{p=1}^n \left(\sum_{i=1}^m z_{t,k,l}^{p,i} \right). \quad (19)$$

G. LOCAL EQUIVALENCE CLASSES

In what follows, we will see that finding a specific operation, such as the CNOT gate, is not only a formidable task but also a frequently unsuccessful one. In the case of two-qubit operations, this amounts to searching for a specific point in the 15-D manifold of the Lie group $SU(4)$.

Since we are primarily interested in operations that can change entanglement between two quantum bits, we can replace a target gate T as the compilation objective by its local equivalence class $[T]$, that is, by all unitary operations that are equivalent to T up to arbitrary single-qubit transformations [28]. This significantly reduces the optimization problem from a search in 15-D manifold of the group $SU(4)$ to 3-D manifold of the Weyl chamber of the local equivalence classes [32]. This in turn can be regarded as a significant increase in the “size” of the optimization target [32].

Particularly useful instruments for defining local equivalence classes are the local invariants defined by Makhlin [42] and further studied in the context of quantum control by Zhang et al. [28], where more details about their construction can be found.

A two-qubit unitary operator $U \in SU(4)$ has the Cartan decomposition

$$U = k_1 A k_2 \quad (20)$$

where k_1 and k_2 are from the single-qubit subgroup $SU(2) \otimes SU(2)$ of $SU(4)$. Importantly, A , which is generated by the Cartan subalgebra of the Lie algebra $SU(4)$, encodes all the nonlocal or entangling properties of the unitary operation.

We can define the local invariants as follows: We first transform the matrix U into the Bell basis

$$U_B = Q^\dagger U Q \quad (21)$$

using the matrix

$$Q = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & i \\ 0 & i & 1 & 0 \\ 0 & i & -1 & 0 \\ 1 & 0 & 0 & -i \end{pmatrix}. \quad (22)$$

This transformation makes the single qubit terms of the Cartan decomposition k_1 and k_2 elements of the special orthogonal group $SO(4)$. The non-local matrix A transforms into a diagonal form. Then, calculating the Makhlin matrix

$$m_U = U_B^T U_B \quad (23)$$

eliminates one of the local components, as the product $O^T O$ is the identity for $O \in SO(4)$. The second of the local operations can be shown to diagonalize the Makhlin matrix m_U . From the characteristic equation and the factorization of an arbitrary unitary matrix from the group $U(4)$ into a product of the global phase $U(1)$ and $SU(4)$, we can derive the local invariants for an arbitrary two-qubit unitary matrix in the following form:

$$\begin{aligned} g_1 &= \text{Re} \left\{ \frac{\text{tr}^2(m_U)}{16 \det(U)} \right\} & g_2 &= \text{Im} \left\{ \frac{\text{tr}^2(m_U)}{16 \det(U)} \right\} \\ g_3 &= \frac{\text{tr}^2(m_U) - \text{tr}(m_U^2)}{4 \det(U)}. \end{aligned} \quad (24)$$

Instead of compiling exactly a target gate, we could instead attempt to compile a gate that belongs to the local equivalence class of the target gate [32]. A suitable “distance” measure to the CNOT local equivalence class can be defined using these invariants, for example, as in [32]

$$D_{\text{CNOT}}(U) = g_1^2 + g_2^2 + (g_3 - 1)^2. \quad (25)$$

In some applications, we may be interested in generating an arbitrary perfect entangler, that is, an operation capable of creating a maximally entangled state out of some product state. This problem can be solved by using the cost function developed in [33]. This cost function can be defined in the following form:

$$D_{\text{PE}}(U) = \left(g_1 - g_3 \sqrt{g_1^2 + g_2^2} \right)^2. \quad (26)$$

H. DISTANCE TO THE LOCAL EQUIVALENCE CLASS MIQCQP PROBLEM FORMULATION

In this section, we elaborate on how to implement the cost function from the previous section as an MIQCQP program. The encoding of (21) and (23) is straightforward and analogous to the encoding of (5). The division $c = a/b$ can be implemented as a bilinear constraint $a = bc$. The determinant can be calculated using a recursive Laplace (cofactor) expansion [43, p. 752, Sect. VI.23] that leads to bilinear constraints. The constraints are as follows: For a given matrix A , we define a variable z_A that is constrained to have the value of $\det(A)$. If A is a 2×2 matrix, then the following bilinear constraint ensures that $z_A = \det(A)$:

$$z_A = A_{11}A_{22} - A_{12}A_{21}. \quad (27)$$

If A is an $n \times n$ matrix, where $n > 2$. Then

$$z_A = \sum_{j=1}^n (-1)^j A_{1j} z_{A,1j} \quad (28)$$

where variable $z_{A,1j}$ is constrained to have a value of the determinant of matrix A with the first row and j th column removed, which can be achieved by recursively adding the constraint (28) for the submatrices of A .

For the compilation of a gate in the CNOT local equivalence class, we use the following cost function:

$$J(Y_d^C) = D_{\text{CNOT}}(Y_d^C). \quad (29)$$

While for the compilation of a general perfect entangler gate, we use the following cost function:

$$J(Y_d^C) = D_{\text{PE}}(Y_d^C). \quad (30)$$

Note that we do not replace any of the constraints used for computing either $D_{\text{CNOT}}(Y_d^C)$ or $D_{\text{PE}}(Y_d^C)$ using McCormick linearization, as this linearization is not exact for bilinear forms where both variables are continuous.

I. EXTRACTING GATE (BRAID) SEQUENCE FROM AN MIQCQP SOLUTION

The result of the compilation problem is a gate (braid) sequence whose product minimizes the cost function (8). This sequence is obtained from a solution of the MIQCQP program II-E as follows. For $t = 1, \dots, d$, the value of the t th element of the sequence is the index i of the nonzero variable $x_{t,i}$.

J. SOLVING THE MIQCQP PROGRAMS

The above MIQCQP programs can be implemented using various free or commercial solvers, such as CPLEX, SCIP, or Gurobi. In our work, we implemented the MIQCQP programs in Gurobi [44]. The Gurobi solver for mixed-integer quadratic programs is based on a branch-and-bound-and-cut framework [45], and employs the simplex method [46], as well as the interior-point and barrier methods [47], [48], to solve the relaxations of linear and quadratic programming.

Note that MIQCQP solvers typically do not operate directly on complex arithmetic. This limitation can be overcome by decomposing all complex expressions into their real and imaginary components and formulating the constraints and objective function in terms of these components.

III. ISING ANYONS FROM A NONSEMISIMPLE TOPOLOGICAL QUANTUM FIELD THEORY

In order to numerically evaluate the proposed method, we will use a framework for topological quantum computation based on the newly discovered nonsemisimple topological quantum field theory in 2+1 dimensions [26]. This theory possesses, besides familiar topological charges of the Ising theory, i.e., trivial charge $\mathbb{1}$, Ising anyons σ , and fermions ψ , an additional topological charge α which gives rise to additional fusion rules. One such rule that is particularly relevant to one and two-qubit encodings is

$$\alpha \times \sigma = (\alpha + 1) + (\alpha - 1). \quad (31)$$

Following [26], the single-qubit computational basis states are encoded via:

$$|0\rangle = ((\alpha, \sigma)_{\alpha+1}, \sigma)_{\alpha} \quad |1\rangle = ((\alpha, \sigma)_{\alpha-1}, \sigma)_{\alpha}. \quad (32)$$

Here, the first fusion produces the topological charge $\alpha + 1$ and $\alpha - 1$, respectively, which is then followed by the second fusion with an Ising anyon, resulting in the total topological charge α for this three-anyon encoding.

In this setting, the quantum computing gates arise from braiding anyons, which is a purely topological operation and does not depend on a time parameter or an angle. In the following, we define the gate alphabet following the notation in [26].

Let q be an eighth root of unity, i.e., q can be expressed as $e^{2\pi i k/8}$ for some $k \in \{0, \dots, 7\}$. In our calculations, we set $k = 1$, and thus $q = e^{2\pi i/8}$. For $\alpha \in \mathbb{R}$, we define the real coefficients $B_{\alpha+1}$ and $B_{\alpha-1}$ by

$$B_{\alpha+1} := \frac{\sqrt{2}}{-1 + \cot \frac{\pi(\alpha+1)}{4}}, \quad B_{\alpha-1} := \frac{\sqrt{2}}{-1 + \cot \frac{\pi\alpha}{4}}. \quad (33)$$

Single-qubit operations result from two braiding operations within the single-qubit encoding. In the first braid, the Ising anyon next to the charge α completely encircles it counter-clockwise. This corresponds to a sequence of two elementary braids or interchanges between both particles, and it gives rise to the matrix $b_1^{(1)2}$, where the upper index (1) indicates a single qubit operation. The second braid operation corresponds to a single elementary braid between the Ising anyons of the encoding, and gives rise to the matrix $b_2^{(1)}$. Both single-qubit matrices are defined via

$$b_1^{(1)2} = -q (b_1^{\alpha\sigma\sigma})^2 = \begin{pmatrix} q^{\alpha} & 0 \\ 0 & q^{-\alpha} \end{pmatrix} \\ b_2^{(1)} = q^{-\frac{3}{2}} b_2^{\alpha\sigma\sigma} = q^{-1} \begin{pmatrix} \frac{1+q^2}{1-q^{2\alpha}} & q^{-1} \frac{\sqrt{B_{\alpha+1}}}{\sqrt{B_{\alpha-1}}} \\ q^{-1} \frac{\sqrt{B_{\alpha+1}}}{\sqrt{B_{\alpha-1}}} & \frac{1+q^2}{1-q^{-2\alpha}} \end{pmatrix}. \quad (34)$$

The matrices $b_1^{\alpha\sigma\sigma}$ and $b_2^{\alpha\sigma\sigma}$ are explicitly

$$(b_1^{\alpha\sigma\sigma})^2 = - \begin{pmatrix} q^{\alpha-1} & 0 \\ 0 & q^{-\alpha-1} \end{pmatrix} \quad (35)$$

$$b_2^{\alpha\sigma\sigma} = q^{\frac{1}{2}} \begin{pmatrix} \frac{1+q^2}{1-q^{2\alpha}} & q^{-1} \frac{\sqrt{B_{\alpha+1}}}{\sqrt{B_{\alpha-1}}} \\ q^{-1} \frac{\sqrt{B_{\alpha+1}}}{\sqrt{B_{\alpha-1}}} & \frac{1+q^2}{1-q^{-2\alpha}} \end{pmatrix}. \quad (36)$$

We now focus on a system of two qubits. Two qubits are encoded into the system of five anyons, consisting of one topological charge α and four Ising anyons. The basis states of the resulting six-dimensional Hilbert space, including four computational states and two noncomputational states, are defined as

$$|00\rangle = (((\alpha, \sigma)_{\alpha+1}, \sigma)_{\alpha}, \sigma)_{\alpha+1}, \sigma)_{\alpha}$$

$$|10\rangle = (((\alpha, \sigma)_{\alpha-1}, \sigma)_{\alpha}, \sigma)_{\alpha+1}, \sigma)_{\alpha}$$

TABLE 1. Results of the numerical experiment for the compilation of the CNOT gate using the gate alphabet NON-SEMI [see (43)]

[illegible]

The value of the MIQCQP cost function $\mathcal{J}(\cdot)$ [see (8)], and the Hilbert–Schmidt distance to CNOT, $d_2(\cdot, \text{CNOT})$, are reported together with the corresponding gate sequence and its length. The results are ordered by $d_2(\cdot, \text{CNOT})$ in increasing order.

$$\begin{aligned}
|01\rangle &= (((\alpha, \sigma)_{\alpha+1}, \sigma)_{\alpha}, \sigma)_{\alpha-1}, \sigma)_{\alpha} \\
|11\rangle &= (((\alpha, \sigma)_{\alpha-1}, \sigma)_{\alpha}, \sigma)_{\alpha-1}, \sigma)_{\alpha} \\
|NC_1\rangle &= (((\alpha, \sigma)_{\alpha+1}, \sigma)_{\alpha+2}, \sigma)_{\alpha+1}, \sigma)_{\alpha} \\
|NC_2\rangle &= (((\alpha, \sigma)_{\alpha-1}, \sigma)_{\alpha-2}, \sigma)_{\alpha-1}, \sigma)_{\alpha}
\end{aligned} \tag{37}$$

Next, we define the braiding of two-qubit systems. Let $\mathbb{1}_2$ be the 2×2 identity matrix. The following two matrices perform the operations on the first qubit of the two-qubit system:

$$b_1^{(2)2} = (b_1^{\alpha\sigma\sigma})^2 \otimes \mathbb{1}_2 \oplus (b_1^{\alpha\sigma\sigma})^2 \quad (38)$$

$$b_2^{(2)} = b_2^{\alpha\sigma\sigma} \otimes \mathbb{1}_2 \oplus \left(q^{\frac{1}{2}} \mathbb{1}_2 \right) \quad (39)$$

where the upper index (2) indicates that these are two-qubit operations, to distinguish the matrices from the single-qubit case [see (34)].

The following two matrices perform the operations on the second qubit of the two-qubit system:

$$J_4 = \mathbb{1}_2 \otimes (b_1^{\alpha\sigma\sigma})^2 \oplus \begin{pmatrix} q^{1-\alpha} & 0 \\ 0 & q^{1+\alpha} \end{pmatrix} \quad (40)$$

$$b_4 = \mathbb{1}_2 \otimes b_2^{\alpha\sigma\sigma} \oplus \left(q^{\frac{1}{2}} \mathbb{1}_2\right). \quad (41)$$

An entangling gate with very low leakage into the non-computational space is constructed in [26] for $\alpha = 2.4$ using an iterative method by Reichardt [49]. Since the leakage term of the constructed entangling gate has a very low leakage term ($\mathcal{O}(10^{-14})$), we will assume that the gate is

leakage-free, in order to simplify our calculations. In the rest of the article, we fix the value of α to 2.4. The entangling gate is a controlled phase shift and is given by

$$\text{CPHASE} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{-1.772i} & 0 \\ 0 & 0 & 0 & e^{-1.682i} \end{bmatrix} \oplus V \quad (42)$$

where V is a unitary 2×2 matrix which acts on the noncomputational subspace NC of the total space which has the structure $C \oplus NC$. Since the matrix V does not affect the computational states, its exact form is irrelevant in our experiments and we choose V to be the 2×2 identity matrix for simplicity.

Finally, the gate alphabet, denoted by NON-SEMI, is a discrete set of fixed unitary operations and is equal to the ordered set $(b_1^{(2)2}, b_2^{(2)}, J_4, b_4, \text{CPHASE})$ where each gate corresponds to the matrices defined in (38)-(42). In the following section, which presents the experimental results (see Tables 1-3), the elements of the gate alphabet NON-SEMI are indexed by the numbers 0 to 4 in the following order:

$$\left(b_1^{(2)2}, b_2^{(2)}, J_4, b_4, \text{CPHASE}\right) \rightarrow (0, 1, 2, 3, 4). \quad (43)$$

IV. NUMERICAL RESULTS

The numerical experiments are divided into six sections as follows. In Section IV-A, we attempt to compile the CNOT gate. In Section IV-B, we search for gates that are as close as possible to the CNOT local equivalence class. In the

TABLE 2. Results of the numerical experiment for the compilation of a gate as close as possible to the local equivalence class of [CNOT] using the gate alphabet NON-SEMI [see (43)]

[illegible]

The value of the MIQCQP cost function $D_{\text{CNOT}}(\cdot)$ [see (29)] is reported together with the corresponding gate sequence and its length. The results are ordered by $D_{\text{CNOT}}(\cdot)$ in increasing order.

TABLE 3. Results of the numerical experiment for the compilation of a perfect entangler gate using the gate alphabet NON-SEMI [see (43)]

[illegible]

The value of the MIQCQP cost function $D_{\text{PE}}(\cdot)$ (Eq.(30)) is reported together with the corresponding gate sequence and its length. The results are ordered by $D_{\text{PE}}(\cdot)$ in increasing order.

Sections IV-C and IV-D, we will broaden our search to the set of perfect entangling gates. In Sections IV-E and IV-F, we demonstrate the universality of our method by applying it to a different gate alphabet, namely, the Fibonacci anyon gate alphabet.

The numerical experiments were run on a computing cluster with AMD EPYC 7543 CPUs. Each experiment was allocated 4 CPU cores and 32 GB of memory. The time limit for the Gurobi solver was set to 23 h, after which we recorded the best solution found, which was not necessarily optimal.

A. CNOT GATE COMPILATION

A natural candidate for the demonstration of our method is the CNOT gate. This gate operates on two qubits and can change the degree of entanglement between them. Since it can also generate a maximally entangled state of two qubits

from some initial non-entangled state, it is also an example of a perfect entangler. In the first set of experiments, we employ the MIQCQP to find a realization of the CNOT gate as a sequence of the operations from the entire gate alphabet NON-SEMI [see (43)]. In accordance with [26], we include the CPHASE operation in the gate alphabet, as defined in (42), which is obtained for the value of the parameter $\alpha = 2.4$. The depth of the circuit, consisting of the elements of the gate alphabet, ranged from 10 to 200 in the optimization process.

The results are shown in Table 1, which lists the best outcomes from this experiment. We reiterate that the gate sequence in the table refers to a sequence of elements of the gate alphabet obtained from a solution of the MIQCQP program as described in Section II-I. We evaluate this experiment using two metrics: 1) the value of the cost function $J(P)$, which is the function that is being minimized

[see (8)], where P is the product of all the gates in the found gate sequence, and 2) the normalized Hilbert-Schmidt distance between the product P and CNOT, $d_2(P, \text{CNOT})$, given by

$$d_2(A, B) = \left\| \frac{A}{\|A\|_2} - \frac{B}{\|B\|_2} \right\|_2. \quad (44)$$

Here, $\|\cdot\|_2$ is the Hilbert-Schmidt ($p = 2$) norm, and each matrix is normalized by its norm. This normalization ensures the distances are bounded between 0 and 2, providing a sense of scale. We note that this measure can return a zero value for proportional matrices due to this normalization.

We observe that, in this numerical experiment, the quantum compilation process has manifestly failed to find the CNOT operation as a circuit of depth up to 200. The native alphabet of NON-SEMI [see (43)] appears unable to reach this gate within 200 steps. CNOT, like any element of the Lie group $U(4)$, factorizes into a trivial $U(1)$ factor and an element of $SU(4)$. To find a specific operation in $SU(4)$ is then equivalent to finding a point in the manifold of fifteen dimensions. A similar situation, albeit in a different anyonic system, was encountered in Ref. [30]. It seems plausible that NON-SEMI, with its limited number of elements, cannot reach the CNOT target in a reasonable number of steps, even though this nonsemisimple anyon system is universal for quantum computation [26].

B. LOCAL CNOT EQUIVALENCE GATE COMPILATION

Here, we shift our focus to the CNOT local equivalence class, that is, all operations which are equivalent to the CNOT gate up to single qubit transformations from the $SU(2) \otimes SU(2)$ subgroup of $SU(4)$. We point out that this significantly reduces the size of the optimization problem from a search in a 15-D space to a search in the three dimensional Weyl chamber. This follows from the fact that each element of $SU(2)$ is represented by three parameters, and an arbitrary element of $SU(2) \otimes SU(2)$ is applied before and after a two-qubit gate, such as CNOT, for example, in order to generate the entire local equivalence class. In other words, by using local invariants, which classify local equivalence classes, we reduce quantum compilation to a 3-D problem. We again use the complete alphabet NON-SEMI [see (43)], which includes the CPHASE operation with α set to 2.4. The depth of the circuits is set from 5 to 80. We use the MIQCQP formulation from Section II-H with the cost function $D_{\text{CNOT}}(\cdot)$, defined by (29).

The results of this numerical experiment are presented in Table 2, which shows the best outcomes in descending order. The best result we were able to obtain comprised of a circuit with depth 35 and accuracy of 10^{-9} , as measured by the $D_{\text{CNOT}}(P)$ distance, where P is the product of the corresponding gate sequence. This measure is significantly below the typical fault-tolerance threshold of 10^{-4} – 10^{-5} (should that ever be needed in this type of system). We note that the best outcome is obtained for circuits that almost exclusively

consist of the repeated application of the CPHASE gate from the gate alphabet.

C. PERFECT ENTANGLER COMPILATION

It is reasonable to consider, in addition to local equivalence classes, the complete set of perfect entanglers as possible optimization targets. These are two-qubit operations capable of creating a maximally entangled state out of some initial product, i.e., a non-entangled state. Specifically, we aim to find an arbitrary gate that is a perfect entangler. We use the entire alphabet NON-SEMI [see (43)], again including CPHASE and α set to 2.4. We will keep the depth in the same range as in the previous case, that is, from 5 to 80. We use the MIQCQP formulation from Section II-H with the cost function $D_{\text{PE}}(\cdot)$, defined by (30).

The results are shown in Table 3, which again shows the best outcomes in descending order with respect to the $D_{\text{PE}}(P)$ distance, where P is the product of the corresponding gate sequence. In particular, the very best results for the circuit depth 77 and 35 exhibit a distance to the perfect entanglers of zero (in standard 64-bit machine precision). The accuracy of the compiled circuits is improved by several orders of magnitude compared to the same distance metrics of the compilation with respect to the local equivalence classes. We again observe that most of the successful compilations are dominated by the repeated application of the CPHASE gate. We will address this point in the next section.

D. GATE COMPILATION USING ONLY THE ENTANGLING GATE

As summarized in Tables 1–3, the MIQCQP quantum compilation found optimal solutions, in which the entangling gate CPHASE features prominently.

This is to be largely expected for the CPHASE gate given by (42). The Mahklin invariants, (24), (g_1, g_2, g_3) turn out to be close to (1,0,3), which characterizes the local equivalence class of the identity, and hence all single qubit operations from $SU(2) \otimes SU(2)$. In fact, in the representation given by the Cartan decomposition, the local equivalence class [CPHASE] lies on the edge of the Weyl chamber [28] that connects the local equivalence classes of the identity [I] and the [CNOT]. Its repeated application then leads to an operation that is closer to the [CNOT] target, eventually reaching it for a sufficiently large number of repetitions.

This is rather remarkable validation of the compilation approach based on the MIQCQP formalism, since the optimal solution it yields agrees with our intuition, which is nevertheless based on sophisticated mathematical formalism (cf., [50]). In particular, Table 3 shows a braiding combination of depth 35, solely consisting of the entangling gate CPHASE, achieving exactly zero distance from the target. In addition to the solution, which clearly captures our intuition, the MIQCQP quantum compilation has also found solutions that would not be accessible in such an intuitive manner.

In order to illustrate the role of CPHASE, we show the distance of successive powers of the entangling gate to the

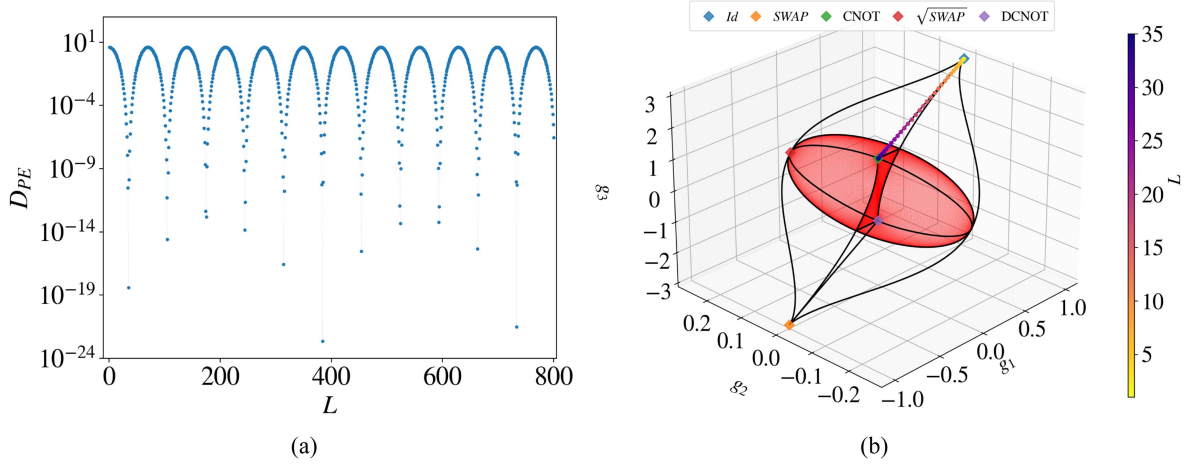


FIGURE 1. (a) Distance of powers of the entangling gate [see (42)], to the perfect entanglers [see (26)] plotted versus power L . (b) Invariants of matrices from panel (a) plotted in g_1, g_2, g_3 space (up to $L = 35$). Data are plotted alongside the perfect entangler section of the Weyl chamber, highlighted in red.

perfect entanglers [see (26)] versus depth or equivalently power L in Figure 1(a). The data highlights the low distances that can be reached with even moderate depth braiding combinations of solely the entangling gate, with a depth of ~ 35 , achieving a distance which is essentially zero in terms of machine precision. In addition, we plot the invariants of circuits with successive powers of CPHASE in Figure 1(b). Here, we also illustrate the Weyl chamber in the invariant space, highlighting the perfect entangler region in red. The relevant equivalence classes are highlighted as points on the vertices of the chamber. This plot illustrates the circuit's path along the edge of the Weyl chamber, starting from approximately the identity class and terminating at the CNOT vertex.

E. CNOT GATE COMPILATION USING FIBONACCI ANYONS GATES

In this section, in order to illustrate the universality of our method, we use a different alphabet of gates. We focus on so-called Fibonacci anyons [8], and use the two-qubit encoding introduced in [25]. The braiding operations are represented by five matrices $\rho_6(\sigma_1), \dots, \rho_6(\sigma_5)$, which are defined as follows. First, defining the matrices R and F

$$R = \text{diag}(R_1^{\tau\tau}, R_\tau^{\tau\tau}) = \begin{pmatrix} e^{-\frac{4\pi i}{5}} & 0 \\ 0 & e^{\frac{3\pi i}{5}} \end{pmatrix} \quad (45)$$

$$F = \begin{pmatrix} \phi^{-1} & \sqrt{\phi^{-1}} \\ \sqrt{\phi^{-1}} & -\phi^{-1} \end{pmatrix}. \quad (46)$$

Then, the braiding matrices are defined as

$$\rho_6(\sigma_1) = R_\tau^{\tau\tau} \oplus (R \otimes \mathbb{1}_2) \quad (47)$$

$$\rho_6(\sigma_2) = R_\tau^{\tau\tau} \oplus (FRF \otimes \mathbb{1}_2) \quad (48)$$

$$\rho_6(\sigma_3) = P_{14}(R_\tau^{\tau\tau} \oplus R \oplus FRF)P_{14} \quad (49)$$

$$\rho_6(\sigma_4) = R_\tau^{\tau\tau} \oplus (\mathbb{1}_2 \otimes FRF) \quad (50)$$

$$\rho_6(\sigma_5) = R_\tau^{\tau\tau} \oplus (\mathbb{1}_2 \otimes R) \quad (51)$$

where P_{14} denotes the permutation matrix that swaps the first and fourth rows of a 5×5 matrix, while $\mathbb{1}_2$ denotes the 2×2 identity matrix.

We denote the gate alphabet $(\rho_6(\sigma_1), \dots, \rho_6(\sigma_5))$ by FIB, and index the elements of FIB in the following order:

$$(\rho_6(\sigma_1), \dots, \rho_6(\sigma_5)) \rightarrow (0, 1, 2, 3, 4). \quad (52)$$

This gate set is known to be universal for quantum computation [51]. The system exists in a 5-dimensional Hilbert space, with a 4-D computational space.

Our goal is to compile a gate of the form $M_{11} \oplus A$, where A is as close to the CNOT gate as possible, and the leakage to noncomputational space is as low as possible. The leakage is indicated by $|M_{11}|$ and $d^U(A)$. The low leakage is represented by the conditions $|M_{11}| \approx 1$ and $d^U(A) \approx 0$. The unitarity measure $d^U(A)$ is defined as $\|A^\dagger A - \mathbb{1}\|_1$.

As discussed in [25] and [30], for this particular encoding, there is a necessary trade-off of leakage for entanglement. Thus, to permit some leakage $\epsilon > 0$, we need to replace constraints (13) and (14) with the following:

$$(L1) \quad \sum_{k,l=n_{NC}+1,\dots,n} \left| \sum_{p=n_{NC}+1}^n [Y_d]_{kp}^* [Y_d]_{pl} - \delta_{kl} \right| < \epsilon \quad (53)$$

$$(L2) \quad \sum_{k,l=1,\dots,n_{NC}} \left| \sum_{p=1}^{n_{NC}} [Y_d]_{kp}^* [Y_d]_{pl} - \delta_{kl} \right| < \epsilon. \quad (54)$$

The constraints (L1) and (L2) imply that $d^U(A) < \epsilon$ and $d^U(M_{11}) < \epsilon$, respectively.

For our numerical experiments, we set ϵ to 0.1, the results of which are reported in Table 4. The compilation of the CNOT gate using Fibonacci anyons was previously studied in [30]. Compared to the random sampling carried out in that work, our method finds significantly better results than in [30].

TABLE 4. Results of the numerical experiment for the compilation of the CNOT gate using the gate alphabet FIB [see (52)]

Depth	$J(\cdot)$	$d_2(\cdot, \text{CNOT})$	$ \mathcal{M}_{11} $	$d^U(\cdot)$	Gate sequence
23	1.4234	0.5998	0.9539	0.0902	34030022213410003301100
21	1.5604	0.6280	0.9539	0.0902	244402304332333334020
16	1.8866	0.6906	0.9539	0.0902	2014433003412443
14	1.8895	0.6911	0.9539	0.0902	43242444234303
18	1.9178	0.6963	0.9538	0.0902	323444432343402340
13	2.1081	0.7300	0.9539	0.0902	4334432224034
11	2.1472	0.7368	0.9539	0.0902	34342224034
19	2.2222	0.7495	0.9539	0.0902	1404440331422203444
15	2.3874	0.7769	0.9539	0.0902	2332100331432324
38	2.4612	0.7888	0.9538	0.0902	14303110341440410310310043204223433433
23	2.5500	0.7984	1.0000	7.3560e-05	03301443004314340043030
17	2.5501	0.7985	1.0000	6.4084e-05	43343000443303004
33	2.5502	0.7985	1.0000	3.8759e-05	0004434004103310300004000040403103
28	2.5855	0.8040	1.0000	1.0921e-04	333333000040000004043000033
38	2.6064	0.8072	1.0000	1.5908e-05	13330000300001100030003101100043340033
22	2.6191	0.8092	1.0000	9.5051e-05	0000000000000000000003
21	2.6192	0.8092	1.0000	4.9379e-05	022444432042302442044
32	2.6192	0.8092	1.0000	1.2882e-04	00000030000000000000000000000000
16	2.6192	0.8092	1.0000	8.6737e-05	0444241044041243
12	2.6193	0.8092	1.0000	4.7810e-05	044444444443

The value of the MIQCQP cost function $\mathcal{J}(\cdot)$ [see (8)], and the Hilbert–Schmidt distance to CNOT, $d_2(\cdot, \text{CNOT})$, are reported together with the corresponding gate sequence, its length, leakage $|M_{\text{fil}}|$, and measure of non-unitarity $d^u(\cdot)$. The results are ordered by $d_2(\cdot, \text{CNOT})$ in increasing order.

TABLE 5 Results of the numerical experiment for the compilation of a gate as close as possible to the local equivalence class of CNOT using the gate alphabet FIB [see (52)]

Depth	$D_{\text{CNOT}}(\cdot)$	$ M_{11} $	$d^U(\cdot)$	Gate sequence
21	1.7083e-05	0.9759	0.0476	220404034041300143224
42	0.4083	0.9718	0.0556	321001404003203240422402014110201121244331
5	2.9065	0.9539	0.0902	12220
6	2.9065	0.9539	0.0902	042224
8	2.9066	0.9539	0.0902	24244442
11	2.9066	0.9539	0.0902	02440200234
20	2.9066	0.9539	0.0902	24310334121244010214
7	2.9066	0.9539	0.0902	4432022
17	2.9066	0.9539	0.0902	24420240444444414
9	2.9066	0.9539	0.0902	213330012

The value of the MIQCP cost function $D_{\text{CNOT}}(\cdot)$ [see (29)] is reported together with the corresponding gate sequence and its length, and the leakage $|M_1|$ and $d^H(\cdot)$. The results are ordered by $D_{\text{CNOT}}(\cdot)$ in increasing order.

F. LOCAL CNOT EQUIVALENCE CLASS GATE COMPI- LATION USING FIBONACCI ANYONS GATES

Here, similarly to Section IV-B, we shift our target from compiling the CNOT gate to instead compiling a gate from the CNOT local equivalence class, but here using the Fibonacci anyon alphabet FIB, (52). We shall use the same constraints set as in the previous section, together with the cost function (29). The results of our numerical experiments are reported in Table 5, in which the leakage tolerance ϵ was set to 0.1. Using our method, we obtain comparable results to those reported in [30].

V. CONCLUSION

We introduced an MIQCQP formulation for quantum circuit compilation. We used McCormick linearization to reduce the number of quadratic constraints. We demonstrated it by compiling the CNOT gate in the nonsemi-simple Ising anyons framework. Then, we provided an MIQCQP formulation for the distance to a local equivalence class and demonstrated it by compiling a gate that is as close as possible to the local equivalence class of the CNOT gate and also to a perfect entangler. Regarding the perfect entangler in a semi-simple Ising anyon framework, the MIQCQP compilation process arrived at the compilation of the perfect entangler given

by a sequence of just one single entangling gate, which is controlled phase rotation and which is native in this anyon system. This is in accord with our intuition regarding controlled phase rotations and the local equivalence classes these operations generate.

REFERENCES

- [1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2000, doi: [10.1017/CBO9780511976667](https://doi.org/10.1017/CBO9780511976667).
- [2] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, vol. 2, Aug. 2018, Art. no. 79, doi: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79).
- [3] P. Shor, "Fault-tolerant quantum computation," in *Proc. 37th Conf. Foundations Comput. Sci.*, 1996, pp. 56–65, doi: [10.1109/SFCS.1996.548464](https://doi.org/10.1109/SFCS.1996.548464).
- [4] R. Knill, Emanuel Laflamme, and W. H. Zurek, "Resilient quantum computing," *Science*, vol. 279, 1998, Art. no.342., doi: [10.1126/science.279.5349.342](https://doi.org/10.1126/science.279.5349.342).
- [5] A. Kitaev, "Fault-tolerant quantum computation by anyons," *Ann. Phys.*, vol. 303, no. 1, pp. 2–30, 2003, doi: [10.1126/science.279.5349.34](https://doi.org/10.1126/science.279.5349.34).
- [6] D. Aharonov and M. Ben-Or, "Fault-tolerant quantum computation with constant error rate," *SIAM J. Comput.*, vol. 38, no. 4, pp. 1207–1282, Jan. 2008, doi: [10.1137/S0097539799359385](https://doi.org/10.1137/S0097539799359385).
- [7] M. H. Freedman, A. Kitaev, M. J. Larsen, and Z. Wang, "Topological quantum computation," *Bull. Amer. Math. Soc.*, vol. 40, 2003, Art. no. 31, doi: [10.1016/S0003-4916\(02\)00018-0](https://doi.org/10.1016/S0003-4916(02)00018-0).
- [8] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, "Non-Abelian anyons and topological quantum computation," *Rev. Mod. Phys.*, vol. 80, pp. 1083–1159, Sep. 2008, doi: [10.1103/RevModPhys.80.1083](https://doi.org/10.1103/RevModPhys.80.1083).

- [9] Z. Wang, *Topological Quantum Computation*. Providence, RI, USA: Amer. Math. Soc., 2010. [Online]. Available: <https://bookstore.ams.org/view?ProductCode=CBMS/112>
- [10] J. K. Pachos, *Introduction to Topological Quantum Computation*. Cambridge, U.K.: Cambridge Univ. Press, 2012, doi: [10.1017/CBO9780511792908](https://doi.org/10.1017/CBO9780511792908).
- [11] S. Xu et al., “Non-Abelian braiding of fibonacci anyons with a superconducting processor,” *Nature Phys.*, vol. 20, no. 9, pp. 1469–1475, 2024, doi: [10.1038/s41567-024-02529-6](https://doi.org/10.1038/s41567-024-02529-6).
- [12] M. Iqbal et al., “Non-Abelian topological order and anyons on a trapped-ion processor,” *Nature*, vol. 626, no. 7999, pp. 505–511, 2024, doi: [10.1038/s41586-023-06934-4](https://doi.org/10.1038/s41586-023-06934-4).
- [13] C. Kassel, O. Dodane, and V. Turaev, *Braid Groups, ser. Graduate Texts in Mathematics* New York, NY, USA: Springer, 2008. [Online]. Available: <https://doi.org/10.1007/978-0-387-68548-9>
- [14] N. E. Bonesteel, L. Hormozi, G. Zikos, and S. H. Simon, “Braid topologies for quantum computation,” *Phys. Rev. Lett.*, vol. 95, Sep. 2005, Art. no. 140503, doi: [10.1103/PhysRevLett.95.140503](https://doi.org/10.1103/PhysRevLett.95.140503).
- [15] S. H. Simon, N. E. Bonesteel, M. H. Freedman, N. Petrovic, and L. Hormozi, “Topological quantum computing with only one mobile quasiparticle,” *Phys. Rev. Lett.*, vol. 96, Feb. 2006, Art. no. 070503, doi: [10.1103/PhysRevLett.96.070503](https://doi.org/10.1103/PhysRevLett.96.070503).
- [16] L. Hormozi, G. Zikos, N. E. Bonesteel, and S. H. Simon, “Topological quantum compiling,” *Phys. Rev. B*, vol. 75, Apr. 2007, Art. no. 165310, doi: [10.1103/PhysRevB.75.165310](https://doi.org/10.1103/PhysRevB.75.165310).
- [17] C. Carnahan, D. Zeuch, and N. E. Bonesteel, “Systematically generated two-qubit anyon braids,” *Phys. Rev. A*, vol. 93, May 2016, Art. no. 052328, doi: [10.1103/PhysRevA.93.052328](https://doi.org/10.1103/PhysRevA.93.052328).
- [18] B. W. Reichardt, “Systematic distillation of composite fibonacci anyons using one mobile quasiparticle,” *Quantum Inf. Computation*, vol. 12, no. 9–10, pp. 876–892, Sep. 2012, doi: [10.26421/QIC12.9-10-10](https://doi.org/10.26421/QIC12.9-10-10).
- [19] V. Kliuchnikov, A. Bocharov, and K. M. Svore, “Asymptotically optimal topological quantum compiling,” *Phys. Rev. Lett.*, vol. 112, Apr. 2014, Art. no. 140504, doi: [10.1103/PhysRevLett.112.140504](https://doi.org/10.1103/PhysRevLett.112.140504).
- [20] M. Burrello, H. Xu, G. Mussardo, and X. Wan, “Topological quantum hashing with the icosahedral group,” *Phys. Rev. Lett.*, vol. 104, Apr. 2010, Art. no. 160502, doi: [10.1103/PhysRevLett.104.160502](https://doi.org/10.1103/PhysRevLett.104.160502).
- [21] Y.-H. Zhang, P.-L. Zheng, Y. Zhang, and D.-L. Deng, “Topological quantum compiling with reinforcement learning,” *Phys. Rev. Lett.*, vol. 125, Oct. 2020, Art. no. 170501, doi: [10.1103/PhysRevLett.125.170501](https://doi.org/10.1103/PhysRevLett.125.170501).
- [22] Q. Chen, Y. Du, Y. Jiao, X. Lu, X. Wu, and Q. Zhao, “Efficient and practical quantum compiler towards multi-qubit systems with deep reinforcement learning,” *Quantum Sci. Technol.*, vol. 9, no. 4, Jul. 2024, Art. no. 045002, doi: [10.1088/2058-9565/ad420a](https://doi.org/10.1088/2058-9565/ad420a).
- [23] H. Xu and X. Wan, “Constructing functional braids for low-leakage topological quantum computing,” *Phys. Rev. A*, vol. 78, Oct. 2008, Art. no. 042325, doi: [10.1103/PhysRevA.78.042325](https://doi.org/10.1103/PhysRevA.78.042325).
- [24] R. Ainsworth and J. K. Slingerland, “Topological qubit design and leakage,” *New J. Phys.*, vol. 13, no. 6, Jun. 2011, Art. no. 065030, doi: [10.1088/1367-2630/13/6/065030](https://doi.org/10.1088/1367-2630/13/6/065030).
- [25] S. X. Cui, K. T. Tian, J. F. Vasquez, Z. Wang, and H. M. Wong, “The search for leakage-free entangling Fibonacci braiding gates,” *J. Phys. A: Math. Theor.*, vol. 52, no. 45, 2019, Art. no. 455301, doi: [10.1088/1751-8121/ab488e](https://doi.org/10.1088/1751-8121/ab488e).
- [26] F. Iulianelli, S. Kim, J. Sussan, and A. D. Lauda, “Universal quantum computation using Ising anyons from a non-semisimple Topological Quantum Field Theory,” *Nature Commun.*, vol. 16, 2025, Art. no. 6408, doi: [10.1038/s41467-025-61342-8](https://doi.org/10.1038/s41467-025-61342-8).
- [27] H. L. Stormer, “Nobel lecture: The fractional quantum Hall effect,” *Rev. Modern Phys.*, vol. 71, no. 4, 1999, Art. no. 875, doi: [10.1103/RevModPhys.71.875](https://doi.org/10.1103/RevModPhys.71.875).
- [28] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, “Geometric theory of non-local two-qubit operations,” *Phys. Rev. A*, vol. 67, 2003, Art. no. 042313, doi: [10.1103/PhysRevA.67.042313](https://doi.org/10.1103/PhysRevA.67.042313).
- [29] Y. Makhlin, “Nonlocal properties of two-qubit gates and mixed states, and the optimization of quantum computations,” *Quantum Inf. Process.*, vol. 1, no. 4, pp. 243–252, 2002, doi: [10.1023/A:1022144002391](https://doi.org/10.1023/A:1022144002391).
- [30] P. C. Burke, C. Aravanis, J. Aspman, J. Mareček, and J. Vala, “Topological quantum compilation of two-qubit gates,” *Phys. Rev. A*, vol. 110, 2024, Art. no. 052616, doi: [10.1103/PhysRevA.110.052616](https://doi.org/10.1103/PhysRevA.110.052616).
- [31] D. DiVincenzo, D. Bacon, J. Kempe, G. Burkard, and K. B. Whaley, “Universal quantum computation with the exchange interaction,” *Nature*, vol. 408, 2000, Art. no. 339, doi: [10.1038/3504254111099036](https://doi.org/10.1038/3504254111099036).
- [32] M. M. Müller, D. M. Reich, M. Murphy, H. Yuan, J. Vala, K. B. Whaley, T. Calarco, and C. P. Koch, “Optimizing entangling quantum gates for physical systems,” *Phys. Rev. A*, vol. 84, 2011, Art. no. 042315, doi: [10.1103/PhysRevA.84.042315](https://doi.org/10.1103/PhysRevA.84.042315).
- [33] P. Watts, J. Vala, M. M. Müller, T. Calarco, K. B. Whaley, D. M. Reich, M. H. Goerz, and C. P. Koch, “Optimizing for an arbitrary perfect entangler. I. Functionals,” *Phys. Rev. A*, vol. 91, 2015, Art. no. 062306, doi: [10.1103/PhysRevA.91.062306](https://doi.org/10.1103/PhysRevA.91.062306).
- [34] M. H. Goerz et al., “Optimizing for an arbitrary perfect entangler. II. Application,” *Phys. Rev. A*, vol. 91, 2015, Art. no. 062307, doi: [10.1103/PhysRevA.91.062307](https://doi.org/10.1103/PhysRevA.91.062307).
- [35] M. Tawarmalani and N. V. Sahinidis, *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming*. New York, NY, USA: Springer, 2002, doi: [10.1007/978-1-4757-3532-1](https://doi.org/10.1007/978-1-4757-3532-1).
- [36] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, New York, USA: Cambridge Univ. Press, 2023, doi: [10.1017/CBO9780511804441](https://doi.org/10.1017/CBO9780511804441).
- [37] S. Burer and A. N. Letchford, “Non-convex mixed-integer nonlinear programming: A survey,” *Surv. Operations Res. Manag. Sci.*, vol. 17, no. 2, pp. 97–106, Jul. 2012, doi: [10.1016/j.sorms.2012.08.001](https://doi.org/10.1016/j.sorms.2012.08.001).
- [38] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, “Mixed-integer nonlinear optimization,” *Acta Numerica*, vol. 22, pp. 1–131, 2013, doi: [10.1016/j.pmcj.2017.07.010](https://doi.org/10.1016/j.pmcj.2017.07.010).
- [39] P. Hruševš and P. Pudlák, “Random formulas, monotone circuits, and interpolation,” in *Proc. IEEE 58th Annu. Symp. Foundations Comput. Sci.*, 2017, pp. 121–131, doi: [10.1109/FOCS.2017.20](https://doi.org/10.1109/FOCS.2017.20).
- [40] M. Gläser and M. E. Pfetsch, “Sub-Exponential Lower Bounds for Branch-and-Bound with General Disjunctions via Interpolation,” in *Proc. 2024 Annual ACM-SIAM Symp. Discrete Algorithms (SODA)*, 2024, pp. 3747–3764, doi: [10.1137/1.9781611977912.132](https://doi.org/10.1137/1.9781611977912.132).
- [41] G. P. McCormick, “Computability of global solutions to factorable non-convex programs: Part I — Convex underestimating problems,” *Math. Program.*, vol. 10, no. 1, pp. 147–175, 1976, doi: [10.1007/BF01580665](https://doi.org/10.1007/BF01580665).
- [42] Y. Makhlin, “Nonlocal properties of two-qubit gates and mixed states and optimization of quantum computations,” *Quantum Inf. Process.*, vol. 1, no. 4, pp. 243–252, 2002, doi: [10.1023/A:1022144002391](https://doi.org/10.1023/A:1022144002391).
- [43] T. Gowers, *The Princeton Companion to Mathematics*. Princeton, NJ, USA: Princeton Univ. press, 2008. [Online]. Available: <https://press.princeton.edu/books/hardcover/9780691118802/the-princeton-companion-to-mathematics>
- [44] Gurobi Optimization, LLC, “Gurobi optimizer reference manual,” 2024. [Online]. Available: https://docs.gurobi.com/_downloads/optimizer/en/13.0/pdf/
- [45] J. E. Mitchell, “Branch and Cut,” in *Wiley Encyclopedia of Operations Research and Management Science*, John Wiley & Sons, Ltd., 2011. [Online]. Available: <https://doi.org/10.1002/9780470400531.eorms0117>
- [46] G. B. Dantzig, “Maximization of a linear function of variables subject to linear inequalities,” RAND Corp., Santa Monica, CA, USA, Tech. Rep. P-38, 1947.
- [47] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica. An Int. J. Combinatorics Theory Comput.*, vol. 4, no. 4, pp. 373–395, 1984, doi: [10.1007/BF02579150](https://doi.org/10.1007/BF02579150).
- [48] J. Nocedal and S. J. Wright, *Numerical Optimization, 2nd ed.* New York, NY, USA: Springer, 2006, doi: [10.1007/978-0-387-40065-5](https://doi.org/10.1007/978-0-387-40065-5).
- [49] B. W. Reichardt, “Systematic distillation of composite Fibonacci anyons using one mobile quasiparticle,” *Quantum Info. Comput.*, pp. 876–892, Sep. 2012.
- [50] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, “Optimal quantum circuit synthesis from controlled-unitary gates,” *Phys. Rev. A*, vol. 69, 2004, Art. no. 042309, doi: [10.1103/PhysRevA.69.042309](https://doi.org/10.1103/PhysRevA.69.042309).
- [51] M. H. Freedman, M. Larsen, and Z. Wang, “A modular functor which is universal for quantum computation,” *Commun. Math. Phys.*, vol. 227, no. 3, pp. 605–622, 2002, doi: [10.1007/s002200200645](https://doi.org/10.1007/s002200200645).