Block encodings of discrete subgroups on a quantum computer

Henry Lamm[1,*] Ying-Ying Li[2,3,†] Jing Shu,[4,5,6,‡] Yi-Lin Wang[2,3,§] and Bin Xu[4,∥]

[1]*Fermi National Accelerator Laboratory, Batavia, Illinois, 60510, USA*

[2]*Interdisciplinary Center for Theoretical Study, University of Science and Technology of China, Hefei, Anhui 230026, China*

[3]*Peng Huanwu Center for Fundamental Theory, Hefei, Anhui 230026, China*

[4]*School of Physics and State Key Laboratory of Nuclear Physics and Technology, Peking University, Beijing 100871, China*

[5]*Center for High Energy Physics, Peking University, Beijing 100871, China*

[6]*Beijing Laser Acceleration Innovation Center, Huairou, Beijing 101400, China*

We introduce a block encoding method for mapping discrete subgroups to qubits on a quantum computer. This method is applicable to general discrete groups, including crystal-like subgroups such as $\mathbb{BI}$ of $SU(2)$ and $\mathbb{V}$ of $SU(3)$. We detail the construction of primitive gates—the inversion gate, the group multiplication gate, the trace gate, and the group Fourier gate—utilizing this encoding method for $\mathbb{BT}$ and for the first time $\mathbb{BI}$ group. We also provide resource estimations to extract the gluon viscosity. The inversion gates for $\mathbb{BT}$ and $\mathbb{BI}$ are benchmarked on the `Baiwang` quantum computer with estimated fidelities of $40^{+5}_{-4}\%$ and $4^{+5}_{-3}\%$, respectively.

I. INTRODUCTION

Gauge symmetries play important roles in quantum field theories, with the $SU(3) \times SU(2) \times U(1)$ gauge symmetry being particularly important as it encapsulates the interactions in the well-established Standard Model for particle physics. Accurate predictions in the strongly coupled regime of these interactions require substantial computational resources. Over the past few decades, Monte Carlo methods in lattice gauge theory (LGT) have flourished, benefitting from advances in supercomputing capabilities and algorithmic innovations. Nevertheless, challenges persist, particularly in scenarios involving dynamic processes such as transport coefficients of the quark-gluon plasma [1–3], parton physics in hadron collisions [4–9], and out-of-equilibrium evolution in the early Universe [10–13] due to sign problems from the complex-valued nature of the Boltzmann sampling weight. Quantum computers offer a promising avenue to circumvent this challenge by enabling real-time simulations within the Hamiltonian formalism [14–18].

The Hilbert space of LGT is infinite-dimensional, requiring digitization methods to facilitate its mapping onto a finite quantum memory (see Sec VI.b of [17] for different digitization methods). These include the loop-string-hadron (LSH) formulations [19–21] where the Hilbert space is built from gauge-invariant operators, digitizations of independent Wilson loops [22–27], qubitization formulations of gauge theories [28–30], and the focus of this paper—the discrete subgroup approximation [31,32]. Understanding and reducing the theoretical errors in the digitization is an area of active research [33–39].

LGT calculations are performed at lattice spacing $a = a(\beta)$ which approaches zero for asymptomatic free theories as the coupling parameter $\beta \to \infty$. To perform extrapolation to the continuous spacetime limit, calculations need to be done in the scaling regime with $\beta > \beta_s$. For the discrete subgroup approximation, gauge links will become "frozen" to the identity when $\beta > \beta_f$, leading to different behaviors from the continuous groups, which makes the discrete subgroup approximations to be only valid in the regime $\beta < \beta_f$. Thus to extrapolate to the correct continuous spacetime limit as the continuous group, the discrete subgroup approximations need to satisfy the condition $\beta_f > \beta_s$. The discrete subgroup approximation has been explored for the Abelian group $U(1)$ [40,41] and $SU(N)$ gauge theories [35,36,42–49], including fermions [50,51]. For the case of $SU(2)$ which we consider in this work, there

*Contact author: hlamm@fnal.gov

†Contact author: yingyingli@ustc.edu.cn

‡Contact author: jshu@pku.edu.cn

§Contact author: wangyilin@mail.ustc.edu.cn

∥Contact author: binxu@pku.edu.cn

TABLE I. Freezing couplings as a function of spatial dimension d, β_f^{d+1}, for crystal-like subgroups of $SU(2)$. For comparison, $\beta_s^{2+1d} = 3$ and $\beta_s^{2+1d} = 2.2$.

G	β_f^{2+1d}	β_f^{3+1d}
$\mathbb{BT}$[a]	3.45(5)	2.25(5)
$\mathbb{BO}$[b]	5.45(5)	3.25(5)
$\mathbb{BI}$	9.65(5)	5.85(5)

[a]Numerical results from [45].
[b]Numerical results from [53].

are three crystal-like subgroups: $\mathbb{BT}$, $\mathbb{BO}$, and $\mathbb{BI}$ with β_f^d shown in Table I. The smallest group, $\mathbb{BT}$ has $\beta_f^{3+1} < \beta_s^{3+1} \sim 2.2$, while the other two subgroup $\mathbb{BI}$, $\beta_f^d \gg \beta_s^d$ as shown in Fig. 1 for both $2+1d$ and $3+1d$ theories. The larger β_f of $\mathbb{BI}$ implies that digitization errors should be smaller in comparison to $\mathbb{BT}$ and $\mathbb{BO}$. Other attempts to increase β_f with modified actions are also being discussed [52].

Though reducing the uncertainties in the extrapolation to the continuum with larger β_f, quantum simulations of larger discrete group will be more complicated. To simulate the discrete subgroup theory on a quantum computer, one
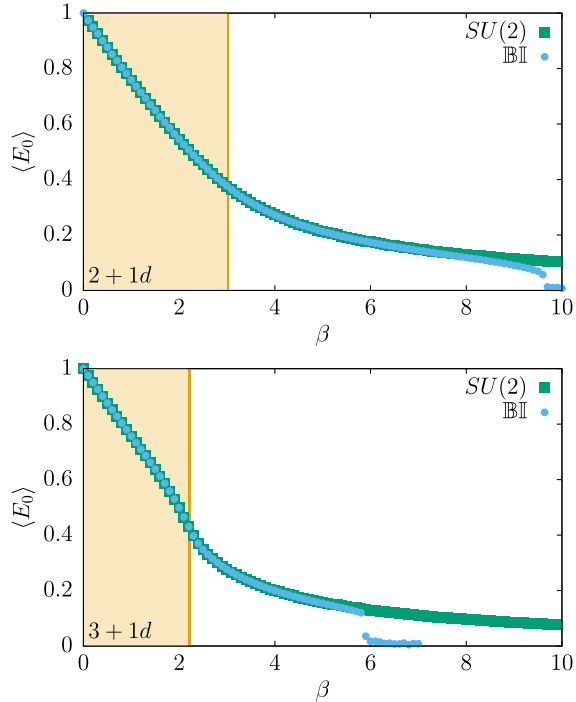

FIG. 1. Average plaquette or lattice energy density $\langle E_0 \rangle$ of $\mathbb{BI}$ as a function of Wilson coupling parameter β on 8^d lattices for (top) $2+1d$ with $\beta_f^{2+1} = 9.65(5)$ and (bottom) $3+1d$ with $\beta_f^{3+1} = 5.85(5)$. The shaded region corresponds to the phase where $\beta \leq \beta_s$.

need to obtain the mapping between the elements of the discrete group and the qubits. The usual way is to write the elements of certain discrete subgroup as an ordered product of group generators with the exponents mapped to qubits or qudits on quantum computer [45,53–55]. However, for the largest crystal-like $\mathbb{BI}$ and $\mathbb{V}$ subgroups of $SU(2)$ and $SU(3)$ theories, the ordered product expression is unknown yet, and may not be possible. This potential obstacle motivates the search for other encodings of discrete groups.

In this work, we consider a new encoding—*block encoding*—expressing groups as d-dimensional matrices over a finite field **F**. According to Cayley's theorem, any finite group of order n can be mapped by an injective homomorphism to the general linear group $GL(d, \mathbf{F})$ over certain finite field **F**, with dimension $d \leq n$. Each matrix element will be sorted in its own register, and then the register representing the group element is built from them. This encoding methods can be applied to generic discrete subgroups, including the largest discrete crystal-like $\mathbb{BI}$ subgroup of $SU(2)$ and $\mathbb{V}$ of $SU(3)$. After reviewing the basics of group representation in Sec. II, we introduce the block encoding methods. In Sec. III, we review the primary gates to implement group element operations and basic logic quantum gates set in constructing the primary gates. We then present the construction of primary gates based on the block encoding for the $\mathbb{BT}$, $\mathbb{BI}$ in Sec. IV. Resource requirements are estimated and compared to other digitization methods in Sec. V. Fidelities for the block encoding with quantum errors are analyzed in Sec. VI, followed by the benchmarking of the inverse gates in Sec. VII. In Sec. VIII, we present the conclusion and outlook. Appendixes include an alternative constructions based on two's complement for block encoding.

II. DISCRETE SUBGROUP ENCODINGS

For any encoding of discrete groups, the group elements g are mapped to at least as many integers as $p = [0, |G|-1]$ where $|G|$ is the group dimension. Previous work focused on the *ordered product* encoding which maps onto the integers the integer exponents $\{o_k\}$ of group generators $\{\lambda_k\}$ in a particular ordering

$$g_{\{o_k\}} = \prod_k \lambda_k^{o_k} \qquad (1)$$

where $0 \leq o_k < O_k$. O_k can be as large as the generator's order $\lambda_k^{O_k} = \mathbb{1}$ but often lower when redundancies occur e.g. $\lambda_1^{O_1} = \lambda_2$. An integer mapping is then defined as

$$p = o_k + O_k(o_{k-1} + O_{k-1}(\cdots + O_2(o_1 + O_1 o_0))). \qquad (2)$$

With this, one may consider encoding p onto quantum memory by decomposing p via Eq. (2) where $\{O_k\}$ are replaced by the dimensionality of the qudits $\{D_k\}$ available. Both $\mathbb{BT}$ and $\mathbb{BO}$ have been formulated in this

way [45,53]. As an example, the group elements of $\mathbb{BT}$ can be represented as

$$g = (-1)^m \mathbf{i}^n \mathbf{j}^o \mathbf{l}^p, \qquad (3)$$

where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the unit quaternions and $\mathbf{l} = -(\mathbf{1} + \mathbf{i} + \mathbf{j} + \mathbf{k})/2$. m, n, o are 3 binary variables, while p is tertiary, and thus can be encoded for example in a ($p = 8$) quoctit and ($p = 3$) qutrit or in 5 qubits with some forbidden states. A similar construction for $\mathbb{BO}$ could be realized with 6 qubits, or a qutrit with either 2 quoctits ($p = 4$) or a qudecasexit ($p = 16$) [54]. The corresponding qubit primitive gates to implement group inverse, product, trace and Fourier transformation has been worked out recently [45,53]. While this encoding can be efficient in quantum memory, it is not a given that all finite groups can be encoding with ordered products.

Due to the potential limitations of the ordered product encoding, we consider a different encoding, block encoding, which instead maps the finite fields in the matrices representing g to a set of qubits. This encoding relies on the ability to represent the matrix elements themselves as valued only over a finite field. Here, we will consider how to encode $\mathbb{BT}$ and $\mathbb{BI}$ as examples.

The $\mathbb{BT}$ group is isomorphic to the special linear group $SL(2,3)$—the group of all 2×2 matrices with unit determinant over the three-element finite field $\mathbf{F}_3$ (which can be defined as the ring of integers modulo 3). The character table for $\mathbb{BT}$ is found at the bottom of Table II. Using the isomorphism, g can be represented by

$$g \in \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \middle| a, b, c, d \in \mathbf{F}_3, ad - bc \equiv 1 \mod (3) \right\} \qquad (4)$$

This leads to another way of encoding $\mathbb{BT}$ group elements using four ternary variables $a, b, c, d = \{0, 1, 2\}$ as $|g\rangle = |abcd\rangle$ with a Hilbert space of $3^4 - 57 = 24$ states where 57 states are removed by the determinant constraints.

In this work, we encode each matrix element in Eq. (4) as a binary integer into two qubits, e.g., the non-negtive encoding with $|0\rangle = |00\rangle$, $|1\rangle = |01\rangle$, $|2\rangle = |10\rangle$ and a forbidden state $|11\rangle$. Given the redundancy in this encoding, we can also use the two's complement encoding as $|0\rangle = |00\rangle$, $|1\rangle = |01\rangle$ and $|2\rangle = |11\rangle$. In the $\mathbb{BT}$ case, the two's component encoding is the same as the gray code [56–59]. Alternatively, the matrix elements could each be encoding in a ($p = 3$) qutrit without forbidden states. In this paper, we adopt the non-negative encoding, resulting in a 8-qubit group register while leaving optimizations of quantum resources among different encoding to the future as considerations to noise is crucial [54].

The block encoding can be extended to other groups. The $\mathbb{BI}$ group is isomorphic to the special linear group $SL(2,5)$—the group of all 2×2 matrices over the finite field $\mathbf{F}_5$ with unit determinant. The character table for $\mathbb{BI}$ is

TABLE II. Character tables of (top) $\mathbb{BT}$ and (bottom) $\mathbb{BI}$ including a representative element in the given conjugacy class, where $\omega = e^{2\pi i/3}$, $\tau = (1 + \sqrt{5})/2$, and $\sigma = (1 - \sqrt{5})/2$.

$\mathbb{BT}$	C_1	C'_1	C_4	C'_4	C''_4	C'''_4	C_6
Order	1	2	3	3	6	6	4
χ_1	1	1	1	1	1	1	1
$\chi_{1'}$	1	1	ω	ω^2	ω	ω^2	1
$\chi_{1''}$	1	1	ω^2	ω	ω^2	ω	1
χ_2	2	-2	-1	-1	1	1	0
$\chi_{2'}$	2	-2	$-\omega$	$-\omega^2$	ω	ω^2	0
$\chi_{2''}$	2	-2	$-\omega^2$	$-\omega$	ω^2	ω	0
χ_3	3	3	0	0	0	0	-1
$\lvert g\rangle$	$\begin{pmatrix}1&0\\0&1\end{pmatrix}$	$\begin{pmatrix}2&0\\0&2\end{pmatrix}$	$\begin{pmatrix}1&1\\0&1\end{pmatrix}$	$\begin{pmatrix}1&2\\0&1\end{pmatrix}$	$\begin{pmatrix}2&2\\0&2\end{pmatrix}$	$\begin{pmatrix}2&1\\0&2\end{pmatrix}$	$\begin{pmatrix}0&1\\2&0\end{pmatrix}$

$\mathbb{BI}$	C_1	C'_1	C_{30}	C_{20}	C'_{20}	C_{12}	C'_{12}	C''_{12}	C'''_{12}
Ord.	1	2	4	3	6	5	10	5	10
χ_1	1	1	1	1	1	1	1	1	1
χ_2	2	-2	0	-1	1	$-\tau$	τ	$-\sigma$	σ
$\chi_{2'}$	2	-2	0	-1	1	$-\sigma$	σ	$-\tau$	τ
χ_3	3	3	-1	0	0	τ	τ	σ	σ
$\chi_{3'}$	3	3	-1	0	0	σ	σ	τ	τ
χ_4	4	4	0	1	1	-1	-1	-1	-1
χ_5	5	5	1	-1	-1	0	0	0	0
$\chi_{4'}$	4	-4	0	1	-1	-1	1	-1	1
χ_6	6	-6	0	0	0	1	-1	1	-1
$\lvert g\rangle$	$\begin{pmatrix}1&0\\0&1\end{pmatrix}$	$\begin{pmatrix}4&0\\0&4\end{pmatrix}$	$\begin{pmatrix}0&1\\4&0\end{pmatrix}$	$\begin{pmatrix}0&1\\4&4\end{pmatrix}$	$\begin{pmatrix}0&4\\4&4\end{pmatrix}$	$\begin{pmatrix}1&1\\0&1\end{pmatrix}$	$\begin{pmatrix}4&4\\0&4\end{pmatrix}$	$\begin{pmatrix}1&2\\0&1\end{pmatrix}$	$\begin{pmatrix}4&3\\0&4\end{pmatrix}$

found at the bottom of Table II and the group element can be represented by

$$g \in \left\{ \begin{pmatrix} a & b \\ c & d \end{pmatrix} \middle| a, b, c, d \in \mathbf{F}_5, ad - bc \equiv 1 \mod (5) \right\} \qquad (5)$$

We can encode the $\mathbb{BI}$ group elements with four quinary variables $a, b, c, d = \{0, 1, 2, 3, 4\}$, and thus the matrix elements can be represented with three qubits with forbidden states $|101\rangle, |110\rangle, |111\rangle$, and the group element with 12 qubits. Alternatively, the matrix elements could be encoding into ($p = 5$) ququints without forbidden states. In this work, we derive the primitive gates of $\mathbb{BI}$ with a 12-qubit group register.

It is worth noting that the largest crystal-like subgroup of $SU(3)$—$\mathbb{V}$—is isomorphic to a subgroup of $GL(3, 4)$ and can be encoded with nine 9 ququarts or 18 qubits, though further investigations are needed to implement the quantum gates that can project the Hilbert space to its $\mathbb{V}$ subspace.

III. PRIMITIVE GATE OVERVIEWS

Quantum circuits for pure gauge theories can be constructed out of a set of primitive gates [60] acting on one or

more group element registers. The necessary gates for simulation and extraction of observables are

   (i) the inverse gate: $\mathcal{U}_{-1}|g\rangle = |g^{-1}\rangle$, which computes in-place the inverse of a group element,

  (ii) the trace gate: $\mathcal{U}_{\mathrm{Tr}}(\theta)|g\rangle = e^{i\theta\mathrm{Re\,Tr}g}|g\rangle$, which introduces a phase based on the trace of a group element in a particular representation and weighted by a coupling θ which can depend on the Hamiltonian and approximation used in time evolution,

 (iii) the left multiplication gate: $\mathcal{U}_{\times}|g_i\rangle|g_j\rangle|\mathrm{anc}\rangle = |g_i\rangle|g_j\rangle|g_ig_j\rangle$. In this work, we consider a novel definition of $\mathcal{U}_{\times}$ which stores the result in an ancilla group register. The right multiplication gate, if desired, can be defined via the $\mathcal{U}_{-1}$, and $\mathcal{U}_{\times}$,

 (iv) the group Fourier gate: $\mathcal{U}_F \sum_g f(g)|g\rangle = \sum_\rho \hat{f}(\rho)_{ij}|\rho, i, j\rangle$ with $\hat{f}$ denoting the Fourier transform of f.

For qubit-based computers, we need 8 qubits to encode a $\mathbb{BT}$ register and 12 qubits for $\mathbb{BI}$. We will present the primitive gates construction using five entangling gates: the two qubit CNOT and SWAP, the three qubit CSWAP (known as the Fredkin gate), and the multicontrolled phase $C^nP(\phi)$ and C^nNOT quantum gates. The special case of C^2NOT is commonly called the Toffoli gate. The first three implement the operations

$$\mathrm{CNOT}|q_1\rangle|q_2\rangle = |q_1\rangle|q_1 \oplus q_2\rangle \tag{6}$$

$$\mathrm{SWAP}|q_1\rangle|q_2\rangle = |q_2\rangle|q_1\rangle \tag{7}$$

$$\mathrm{CSWAP}|q_1\rangle|q_2\rangle|q_3\rangle = |q_1\rangle|q_3\rangle|q_2\rangle. \tag{8}$$

For the multicontrolled gates, which apply and operation to one qubit based on the state of $n-1$ others we have

$$C^nP(\phi)|q_1\rangle...|q_n\rangle = |q_1\rangle...e^{i\phi q_1...q_{n-1}}|q_n\rangle \tag{9}$$

$$C^n\mathrm{NOT}|q_1\rangle...|q_n\rangle = |q_1\rangle...|q_n \oplus q_1...q_{n-1}\rangle. \tag{10}$$

IV. PRIMITIVE GATES FOR $\mathbb{BT}$ AND $\mathbb{BI}$

To construct the primitive gates based on block encodings, we adopt the following conventions. We use parentheses to denote operations that return values with mod n applied for $SL(2, n)$ group. The values of matrix element are the remainders mod n. A feature of the block encoding is that the primitive gates for $\mathbb{BT}$ and $\mathbb{BI}$ are quite similar, with the difference mainly arising from the modular arithmetic. Therefore we will consider the two groups in parallel as we construct their gates.

A. Inverse gate

For the construction of $\mathcal{U}_{-1}$, we note that the inverse g^{-1} is given in terms of the matrix elements of g by

$$g^{-1} = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}, \tag{11}$$

From this, we see that the inverse operation corresponds to swapping the values of a, d and flipping the sign of b, c. Using this, the $\mathcal{U}_{-1}$ circuits can be derived and are presented in the left of Figs. 2 and 3 for $\mathbb{BT}$ and $\mathbb{BI}$ respectively.

B. Trace gate

Our interest here is in the traces of each group elements in the faithful representation, which are denoted by χ_2 in Table II. With one representative element in the given conjugacy class in Table II, one can determine the other elements in the class. This enables a derivation of the rules for obtaining $\mathrm{Tr}(g)$ in the blocking encoding. For $\mathbb{BT}$, these rules are

$$\mathrm{Tr}(g) = \begin{cases} 2 & \text{if } (a+d) = 2 \ \& \ (c-b) = 0 \\ 1 & \text{if } (a+d) = 1 \ \& \ (c-b) \neq 0 \\ 0 & \text{if } (a+d) = 0 \\ -1 & \text{if } (a+d) = 2 \ \& \ (c-b) \neq 0 \\ -2 & \text{if } (a+d) = 1 \ \& \ (c-b) = 0 \end{cases} \tag{12}$$
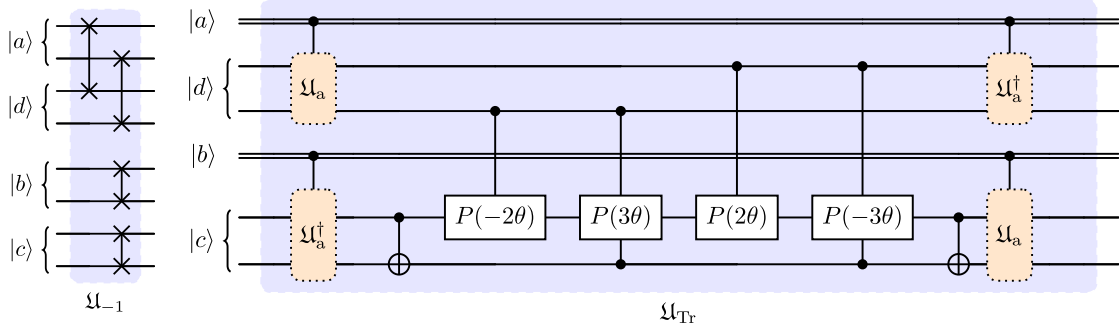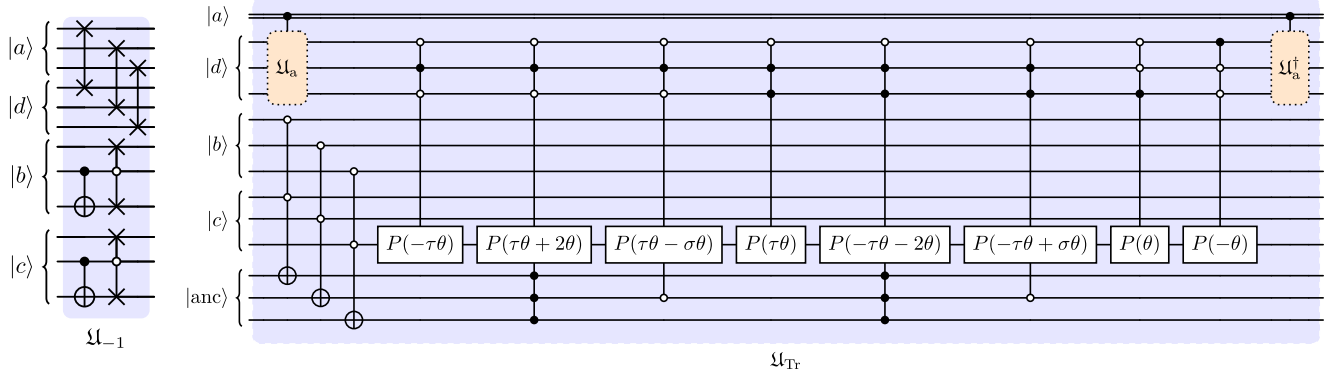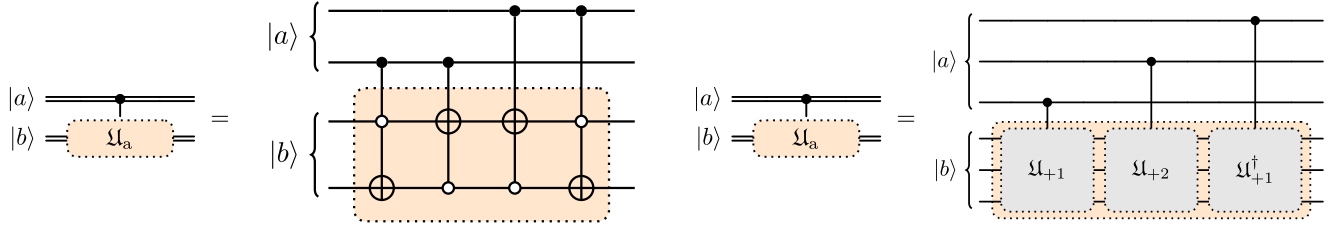

FIG. 2. Inverse gate $\mathcal{U}_{-1}$ (left) and trace gate $\mathcal{U}_{\mathrm{Tr}}(\theta)$ (right) for $\mathbb{BT}$ group.

FIG. 3. Inverse gate $\mathfrak{U}_{-1}$ (left) and trace gate $\mathfrak{U}_{\mathrm{Tr}}$ (right) for $\mathbb{BI}$ group.

FIG. 4. Implementation of $\mathfrak{U}_a$ for $\mathbb{BT}$ (left) and $\mathbb{BI}$ group (right).

While for $\mathbb{BI}$, they are found to be

$$
\mathrm{Tr}(g) = \begin{cases}
0 & \text{if } (a+d) = 0 \\
1 & \text{if } (a+d) = 1 \\
-1 & \text{if } (a+d) = 4 \\
2 & \text{if } (a+d) = 2 \,\&\, (c^2+b^2) = 0 \\
-\tau & \text{if } (a+d) = 2 \,\&\, (c^2+b^2) = 1 \text{ or } 2 \\
-\sigma & \text{if } (a+d) = 2 \,\&\, (c^2+b^2) = 3 \text{ or } 4 \\
-2 & \text{if } (a+d) = 3 \,\&\, (c^2+b^2) = 0 \\
\tau & \text{if } (a+d) = 3 \,\&\, (c^2+b^2) = 1 \text{ or } 2 \\
\sigma & \text{if } (a+d) = 3 \,\&\, (c^2+b^2) = 3 \text{ or } 4.
\end{cases}
\tag{13}
$$

By inspecting these rules, one notices that to realize $\mathfrak{U}_{\mathrm{Tr}}(\theta)$, we first implement an addition operation $\mathfrak{U}_a|a\rangle|b\rangle = |a\rangle|b \oplus a\rangle$. These are constructed for $\mathbb{BT}$ and $\mathbb{BI}$ in Fig. 4. In the case of $\mathbb{BI}$, it is useful to further define subroutines $\mathfrak{U}_{+n}$ for $n = 1$, 2 which increment the matrix element $|a\rangle$ to $|a+n\rangle$ as shown in Fig. 5. The operation of $(b-a)$ can be implemented using $\mathfrak{U}_a^\dagger$.
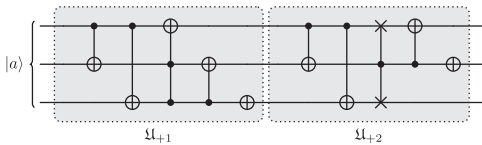
FIG. 5. Elementary operations $\mathfrak{U}_{+1}$ and $\mathfrak{U}_{+2}$ that add one and two to a single matrix element, respectively for the $\mathbb{BI}$ group.

With $\mathfrak{U}_a$, we can construct $\mathfrak{U}_{\mathrm{Tr}}(\theta)$, as shown in the right of Figs. 2 and 3 where θ depends on couplings and approximations in the time evolution.
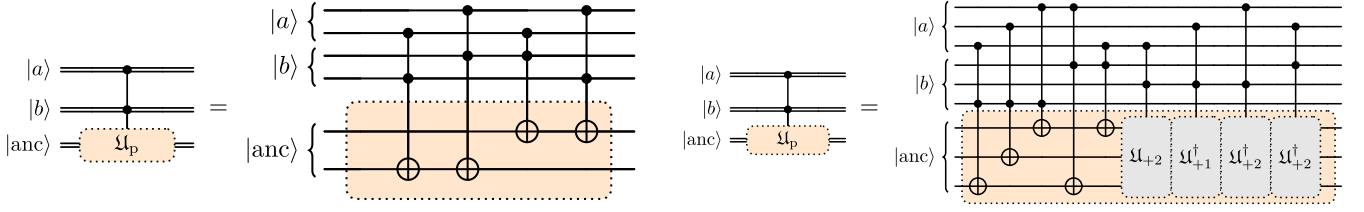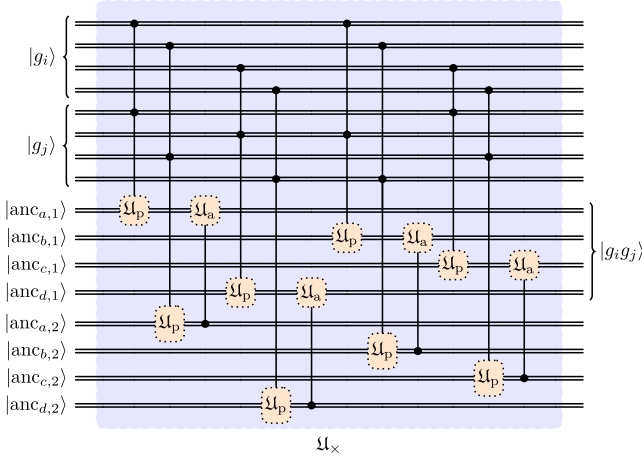
C. Multiplication gate

Moving on to $\mathfrak{U}_\times$, the multiplication of two group elements g_i and g_j in terms of the matrix elements is

$$
\begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix} \begin{pmatrix} a_j & b_j \\ c_j & d_j \end{pmatrix} = \begin{pmatrix} a_i a_j + b_i c_j & a_i b_j + b_i d_j \\ c_i a_j + d_i c_j & c_i b_j + d_i d_j \end{pmatrix}
\tag{14}
$$

The multiplication of group elements can thus be built from blocks of $\mathfrak{U}_a$ and a new subroutine $\mathfrak{U}_p$ which calculate the product of two elements (See Fig. 6). In order to leave the matrix elements intact, ancillary qubits store the product. One ancillary register will be required to store each matrix element in Eq. (14). Nevertheless, to optimize the parallelization, we actually introduce one ancillary register for each product in Eq. (14), which doubles the ancillary qubits required. The group multiplication circuit $\mathfrak{U}_\times$ is subsequently constructed as in Fig. 7.

D. Fourier gate

To reduce the quantum gates required for implementing Fourier gate, we can project both $\mathbb{BT}$ and $\mathbb{BI}$ group to the subspace satisfying the determinate constraints using circuit $\mathfrak{U}_{\mathrm{proj}}$ and ancillary qubits, which will reduce the encoding from 8 and 12 qubits to 6 and 9 qubits, respectively.

FIG. 6. Implementation of $\mathfrak{U}_p$ for $\mathbb{BT}$ (left) and $\mathbb{BI}$ group (right).

FIG. 7. Multiplication gate $\mathfrak{U}_\times$ for $\mathbb{BT}$ group and $\mathbb{BI}$ group.

$\mathfrak{U}_{\text{proj}}$ is implemented with 4 CNOT gates, 3 CSWAP gates and 8 Tofolli gates for $\mathbb{BT}$ group. For the $\mathbb{BI}$ group, $\mathfrak{U}_{\text{proj}}$ requires 8 CNOT gates, 3 CSWAP gates, 8 Toffoli gates and 2 $\mathfrak{U}_p$ gates. With such projection, transforming to the Fourier basis can be decomposed using the QISKIT transpiler. The resource required for Fourier gate implemented this way for both groups are shown in Table III.

$\mathfrak{U}_F$ is usually the most costly primitive gate for quantum simulations. Future direction should consider the fast Fourier transformation in [61] where subexponential growth $\exp(\sqrt{\log|\mathcal{G}|})$ of the circuit depth in the group size is possible. In particular, it should be noticed that the Fourier transformation of $\mathbb{BI}$ group can be built upon that of $\mathbb{BT}$ group using the following natural tower:

$$\mathbb{BI} > \mathbb{BT} > \mathbb{Z}_6 > \mathbb{Z}_3 > \mathbb{Z}_1 = \{1\}. \qquad (15)$$

which could simplify the constructions for the $\mathbb{BI}$ group once the construction for $\mathbb{BT}$ is realized. It will be valuable

to realize such fast Fourier transformation and compare the resources required with our brutal direct decomposition, which we will leave for the future work.

With these four primitive gates, it is possible to perform a resource estimate and compare to other implementations of $SU(2)$. This will be done in Sec. V. Additionally, the primitive gates for the two's component encoding are discussed in the Appendix. There we find that total gate costs are similar, but given the different distribution of qubit states, they are anticipated to have different noise robustness.

V. RESOURCE REQUIREMENTS

As T gate counts are known to require costly encodings for error correction, it is usually used in fault-tolerant resource analysis [62,63]. We estimate the T gates required for simulating both $\mathbb{BT}$ or $\mathbb{BI}$ group with the block encoding method. The Toffoli gate requires 7 T gates [62]; C^nNOT can be constructed with $(2[\log_2(n+1)]-1)$ toffoli gates and $n-2$ dirty ancilla qubits for $n > 2$ [62]. C^nSWAP require the same number of T gates as C^{n+1}NOT, as it can be decomposed to C^{n+1}NOT and CNOTs using the symmetric decomposition [64]. Approximating R_Z gates to the precision of ϵ will require $1.15\log_2(1/\epsilon)$ T gates [53], while R_X and R_Y can be implemented with at most three R_Z gates. $CP(\phi)$ requires 8 T gate and one R_Z gate with one clean ancillary qubit [65], and $C^nP(\theta)$ can be decomposed to $2(n-1)$ Toffoli gates and one $CP(\theta)$ with $n-1$ clean ancilla [66]. The estimated number of T gates to realize each primary gates are listed in Table IV.

We compare the resources required for the calculation of the viscosity as that in [67]. The total T gate count for certain Hamiltonian H is given by $N_T^H = C_T^H \times dL^d N_t$ for a d spatial lattice simulated for a time $t = N_t\delta t$ [53], where C_T^H is the average number of T gates require per link per δt, and dL^d is the total number of links. We first consider the

TABLE III. $\mathfrak{U}_F$ decompositions for $\mathbb{BT}$ and $\mathbb{BI}$ group.

| | $\mathfrak{U}_F$ | | | |
	R_X	R_Y	R_Z	CNOT
$\mathbb{BT}$	186	2052	3491	1941
$\mathbb{BI}$	10743	131879	223044	125919

TABLE IV. Number of T gates required to realize the primary gates.

Gate	T ($\mathbb{BT}$)	T ($\mathbb{BI}$)
$\mathfrak{U}_{-1}$	0	14
$\mathfrak{U}_{\text{Tr}}$	$172 + 4.6\log_2(1/\epsilon)$	$666 + 9.2\log_2(1/\epsilon)$
$\mathfrak{U}_\times$	336	3640
$\mathfrak{U}_F$	$11735.8\log_2(1/\epsilon)$	$748547\log_2(1/\epsilon)$

simulation of the $\mathbb{BT}$ group. With the primitive gates per link per δt listed in [53], we get

$$C_T^{H_{\text{KS}}} = 2102(d-1) + (23469.2 + 2.3d)\log_2 \frac{1}{\epsilon} \quad (16)$$

$$C_T^{H_I} = 8994d - 7650 + (46936.1 + 6.9d)\log_2 \frac{1}{\epsilon}, \quad (17)$$

using the Kogut-Susskind Hamiltonian H_{KS} and the improved Hamiltonian H_I studied in [68], respectively. With this, the total synthesis error ϵ_T can be estimated as the sum of ϵ from each R_Z:

$$\epsilon_T^{H_{\text{KS}}} = 2(10204 + d)dL^d N_t \times \epsilon \quad (18)$$

$$\epsilon_T^{H_I} = 2(20407 + 3d)dL^d N_t \times \epsilon. \quad (19)$$

For simulating the $\mathbb{BI}$, the cost increase moderately:

$$C_T^{H_{\text{KS}}} = 22215(d-1) + (1497090 + 4.6d)\log_2 \frac{1}{\epsilon} \quad (20)$$

$$C_T^{H_I} = 95793d - 81205 + (2994170 + 13.8d)\log_2 \frac{1}{\epsilon}, \quad (21)$$

and total synthesis errors:

$$\epsilon_T^{H_{\text{KS}}} = 4(325454 + d)dL^d N_t \times \epsilon \quad (22)$$

$$\epsilon_T^{H_I} = 12(216969 + d)dL^d N_t \times \epsilon. \quad (23)$$

To calculate the shear viscosity with the total synthesis error $\epsilon_T = 10^{-8}$ on a $d = 3$ lattice with $L^3 = 10^3$ for $N_t = 50$, finding $N_T^{H_{\text{KS}}} = 2.1 \times 10^{11}$ and $N_{T'}^{H_I} = 4.2 \times 10^{11}$. For the ordered product method in simulating $\mathbb{BT}$ group [45], the total number of T gates are estimated as 1.1×10^{11} for H_{KS} and 4.1×10^{11} for H_I. We note that H_{KS} is inadequate to reach the scaling regime with $\mathbb{BT}$, but for the case of H_I, the T gate cost is only $\sim$3% higher—and in both cases dominated by $\mathfrak{U}_F$. With only 60% higher qubit costs, we conclude that this encoding is comparable to ordered product on these simple metrics, and further analysis of other metrics like noise robustness should be undertaken. For the larger $\mathbb{BI}$ group—where no ordered product encoding currently exists—the costs increase to $N_T^{H_{\text{KS}}} = 1.4 \times 10^{13}$ and $N_{T'}^{H_I} = 2.9 \times 10^{13}$ T gates using H_{KS} and H_I, respectively. But given $\beta_f \gg \beta_s$ for $\mathbb{BI}$, the increase in gate costs may be an acceptable trade-off due to the reduced systematic error from digitization.

We can also compare the block encoding of discrete subgroups to other digitization methods, such as LSH formalism, in which the T gate counts has been estimated for $d = 1$ in [69]. For comparison, we take the finite arithmetic precision errors to be smaller than the synthesis error $\epsilon_T = 10^{-8}$ by taking minimally $n = 7$ steps and $m = 42$ bits to evaluate the inverse-square-root functions using

Newton's method. We choose 5 and 7 qubits to encode one group register in LSH to match the minimal number of qubits that can encode $\mathbb{BT}$ and $\mathbb{BI}$ group, respectively. With $L = 10$, LSH will require in trotter step using singular-value-decomposition roughly 3.9×10^7 and 4.2×10^7 T gates for the 5 qubits and 7 qubits case, respectively, while for the blocking encoding, $\mathbb{BT}$ and $\mathbb{BI}$ requires 1.3×10^7 and 8.0×10^8 T gates. Given this similarity in resource costs, quantifying the systematic and statistical errors in digitization methods becomes an important concern. In the next section, we consider the relative resilience of the block encoding versus the ordered producted encoding. Additionally, a comprehensive comparision to other forms of approximating the time evolution in LGT [33,70] are desirable, but understanding the theoretical LGT errors must be understood for a complete comparison.

VI. RESILIENCE TO QUANTUM ERRORS

In this section, we investigate the resilience to errors for a single register using the block encoding and the ordered product encoding, considering $\mathbb{BT}$. Given that redundant degrees of freedom are introduced in both encodings, one can utilize these redundancies in QEC [54,71–82]. In the following, we derive the error rate threshold [82] below which the block encoding would provide higher fidelity than the ordered product one for a single register as more redundancies are introduced in the block encoding. We will take the bit-flip error as an example and consider the fidelity as the one averaging over all group elements. Error mitigation using post-selection remove trials affected by detectable errors. For the error channel with only one bit-flip error $\mathcal{N}_i$ affecting qubit i at an error rate of ϵ, we count the number of $\mathcal{N}_i$ that transform the group element into forbidden state, which are detectable errors. Using the ordered product encoding, there are in total 16 detectable $\mathcal{N}_i$ for all 24 group elements, while for the block encoding, 160 $\mathcal{N}_i$. The lower bound for the fidelity is the probability of no errors after removing trials affected by these detectable one bit-flip errors

$$\mathcal{F}_{\text{BE}}^{\text{ps}} \geq \frac{(1-\epsilon)^8}{1 - 160/24(1-\epsilon)^7 \epsilon},$$

$$\mathcal{F}_{\text{OPE}}^{\text{ps}} \geq \frac{(1-\epsilon)^5}{1 - 16/24(1-\epsilon)^4 \epsilon}. \quad (24)$$

For logical error rates $\epsilon \lesssim 0.1$, we found the lower bound of the fidelity for the current encoding methods can be higher. As postselections require resources that are exponential [83] in system size, we also consider correcting one bit-flip error to reduce the effects of quantum errors. The correctable one bit-flip errors transform the group element to a forbidden state that cannot be transformed from other group element via any one bit-flip error (Knill-Laflamme condition [84]). Under this condition, when a forbidden state is seen, the error channel can be inferred and corrected. Using the ordered
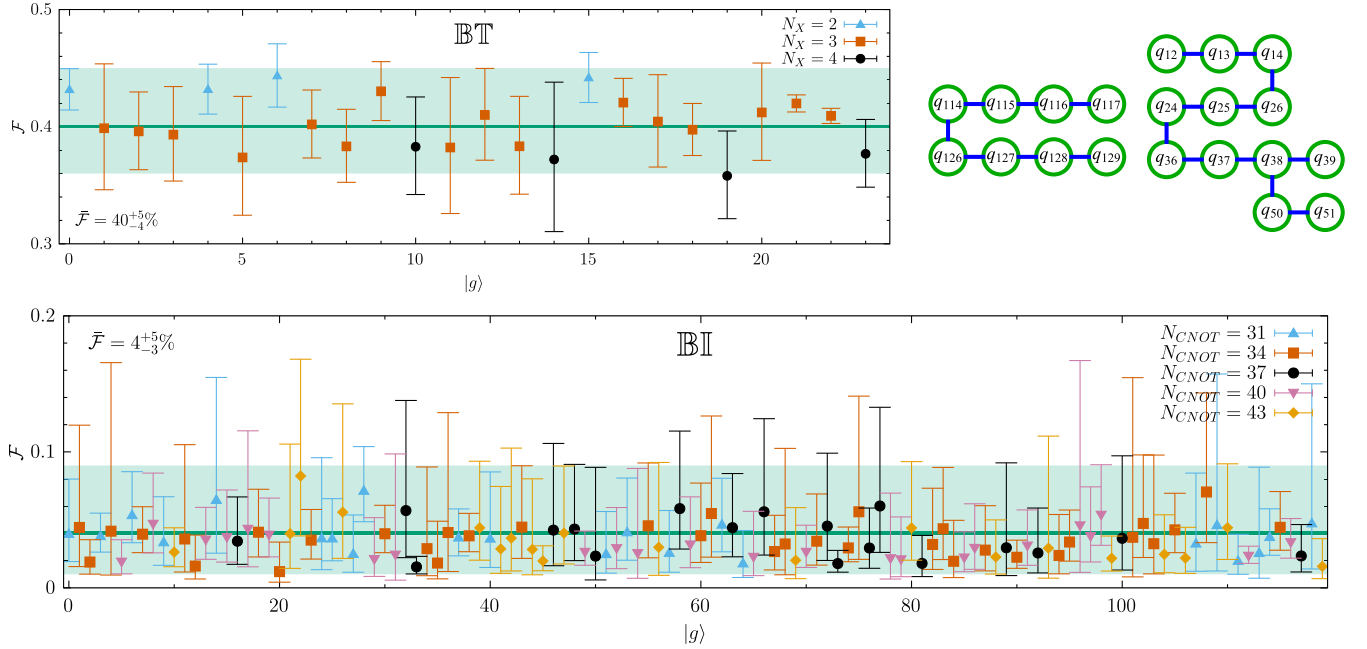
FIG. 8. Fidelity of $\mathfrak{U}_{-1}^{\text{trans}}$ gate of `Baiwang` for each group element $|g\rangle = |abcd\rangle$ of $\mathbb{BT}$ and $\mathbb{BI}$, which is labeled by the lexicographic order in the range $\{0, |G| - 1\}$. The averaged fidelity $\bar{\mathcal{F}}$ over all group elements are also shown. On the top right, the qubit graph on `Baiwang` used to represent $|g\rangle$ for (left) $\mathbb{BT}$ and (right) $\mathbb{BI}$ respectively.

product encoding, there are in total 0 correctable $\mathcal{N}_i$ for all 24 group elements, while for the block encoding, 80 $\mathcal{N}_i$. Correcting trials affected by correctable $\mathcal{N}_i$, we obtain the lower bound of the averaged fidelity as the probability with no errors or only correctable one bit-flip errors

$$\mathcal{F}_{\text{BE}}^{\text{cor}} \geq (1 - \epsilon)^8 + \frac{80}{24}(1 - \epsilon)^7 \epsilon,$$

$$\mathcal{F}_{\text{OPE}}^{\text{cor}} \geq (1 - \epsilon)^5. \qquad (25)$$

We observe that in this situation for $\epsilon \lesssim 0.1$, the lower bound of the fidelity for the current encoding methods can be mildly higher.

VII. EXPERIMENTAL RESULTS

Using the block encoding method, we benchmark the fidelity of the inverse gate for $\mathbb{BT}$ and $\mathbb{BI}$ group on the Quafu platform [85]. Given the availability, we used the `Baiwang` quantum real machine on Quafu which has 144 qubits and 136 available. The qubits are arranged in a 12×12 lattice, where qubits in each row are connected adjacently and adjacent rows are connected by only 2-4 qubit connections [85]. 8 qubits and 12 qubits are selected for the simulation of $\mathbb{BT}$ and $\mathbb{BI}$ group, respectively, with their positions and connectivity shown in Fig. 8. Starting from state $|\psi_0\rangle = |0\rangle^{\otimes n_q}$ with $n_q = 8(12)$ for $\mathbb{BT}(\mathbb{BI})$ group, we can prepare the initial state $|g\rangle$ with circuit $\mathfrak{U}_g: |g\rangle = \mathfrak{U}_g|\psi_0\rangle$. For single states, $\mathfrak{U}_g$ is simply a tensor product of X gates used to initialize some qubits to $|1\rangle$. Adapting to the connectivity of the qubits chosen on `Baiwang` quantum chip, $\mathfrak{U}_{-1}$ requiring

12 (29) CNOT gates[1]) is transpiled to $\mathfrak{U}_{-1}^{\text{trans}}$ with 18 (31–43) CNOTs for $\mathbb{BT}(\mathbb{BI})$ where the number of CNOTs is counted from the transpiled circuits. We subsequently apply $\mathfrak{U}_{-1}^{\text{trans}}$ to $|g\rangle$, resulting in the final state $|g'\rangle$. The fidelity is defined as

$$\mathcal{F} = |\langle g^{-1}|g'\rangle|^2 = |\langle g^{-1}|\mathfrak{U}_{-1}^{\text{trans}}\mathfrak{U}_g|\psi_0\rangle|^2, \qquad (26)$$

which is the probability of measuring the correct $|g^{-1}\rangle$.

We show the fidelity for each g in Fig. 8 for both $\mathbb{BT}$ and $\mathbb{BI}$ group calculated from 5 runs, with $N = 5000$ shots for each run. The group element are labeled by the lexicographic order in the range $\{0, |G| - 1\}$ on the x-axis. We also present the average fidelity $\bar{\mathcal{F}}$ over all group elements in Fig. 8. The fidelity of an X gate is found to be ~97% on `Baiwang`, and CNOT gate to be around ~95%. For the $\mathbb{BT}$ case, different X gates are needed to prepare state, which are denoted with different shapes in the upper plot of Fig. 8. This variance in state preparation causes noticeable variations in $\mathcal{F}$ observed in Fig. 8.

We found that for $|g\rangle$ in $\mathbb{BT}$, the error rate grows slightly with number of X gates in $\mathfrak{U}_{-1}^{\text{trans}}$. We can achieve $\bar{\mathcal{F}} = 40_{-4}^{+5}\%$ for $\mathbb{BT}$. These results, using 8 qubits and 18 CNOTs, can be compared to previous ones from a 5-qubit ordered product encoding of $\mathfrak{U}_{-1}$ [45]. In that work, the transmon-based `ibm_nairobi` was used, which had higher connectivity and Pauli twirling was used for error mitigation. While fewer qubits were required, the order produce

---

[1]We have decomposed the Fredkin gate to 7 CNOTs [64].

encoding required ~ 30 CNOTs. Despite these striking differences, a fidelity of $\bar{\mathcal{F}} = 37.0^{+8}_{-8}\%$ was achieved. This suggests that simplistic comparisons of digitizations on qubit and gate counts alone are insufficient and that factors related to available quantum hardware could be decisive in the search for efficient digitizations.

For $\mathbb{BI}$, $\mathfrak{U}^{\text{trans}}_{-1}$ can require varying numbers of CNOTs depending on the transpilation, including the orders of mapping matrix elements to the qubits. We show the fidelities for each group element of $\mathbb{BI}$ in Fig. 8 (bottom) with shapes indicating the number of CNOTs in the transpiled circuits. As more than 30 CNOT gates are involved, noises from CNOT gates contribute mostly to the quantum errors which reduce the average fidelity to only $\bar{\mathcal{F}} = 4\%$. This low fidelity could be improved by improving transpilation that reduces CNOT counts and implementing error mitigation strategies such as Pauli twirling which have proven effective in the past for LGT [45,68,86–90].

VIII. CONCLUSIONS AND OUTLOOK

In this work, we introduced the block encoding method—the general method for digitizing discrete groups on quantum computers and developed the primitive gate set for the two important discrete subgroup of $SU(2)$, including the first implementation ever of $\mathbb{BI}$. The realization of quantum circuits for this largest crystal-like subgroup of $SU(2)$ allows simulating physics of $SU(2)$ deep into the scaling regime. Applying this digitization to a smaller subgroup of $SU(2)$ within which other digitizations are also feasible, we have shown that the qubit and T costs as well as robustness to noise are comparable. By comparing experimental demonstrations of $\mathfrak{U}_{-1}$ on quantum devices for different encodings, we found that naive comparisons between digitizations based on qubits and gates alone are insufficient, and that true performance on a given quantum architecture must be taken into account. Given this observation about the importance of hardware, block encodings might favor qudit-based quantum computer, and its quantum algorithms deserve future explorations.

A number of directions for research exist following these results. Primary is that given its predominance in the gate costs, determination of a quantum Fourier transformation gate within the block encoding method would be invaluable, as it could radically improve the resources costs. Further, given the block encoding method separates group registers into registers of finite fields, it may greatly benefit from formulations on qudit-based devices similar to [9,55,91–98]. Finally, the outstanding goal of the discrete group approximation is the largest crystal-like subgroup of $SU(3)$—$\mathbb{V}$—which similar to $\mathbb{BI}$ faces obstacles in the order product encoding. Given it is isomorphic to a subgroup of $GL(3, 4)$, the block encoding method provides an avenue for encoding it onto nine 9 ququarts or 18 qubits.

APPENDIX: PRIMITIVE GATES FOR $\mathbb{BT}$ GROUP WITH TWO'S COMPONENT ENCODING

In this section, we present the primitive gates for $\mathbb{BT}$ group with two's component encoding, e.g. $|a = 0\rangle = |00\rangle, |a = 1\rangle = |01\rangle$ and $|a = -1\rangle = |11\rangle$. Given the
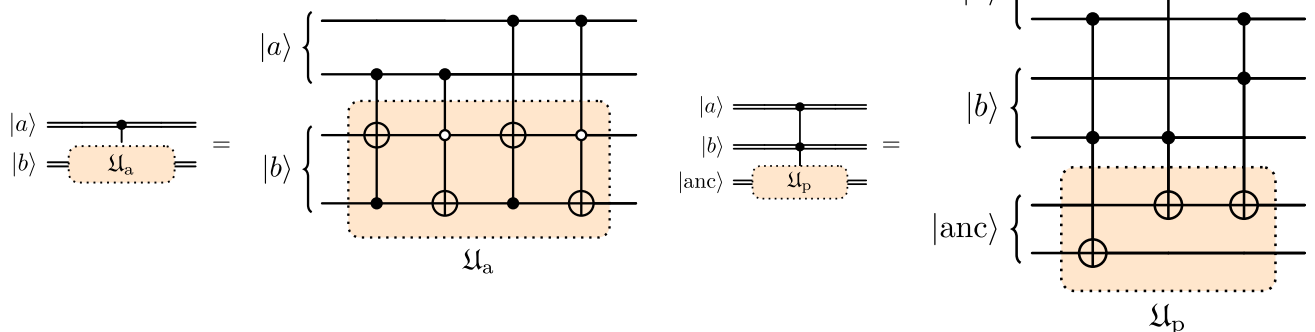

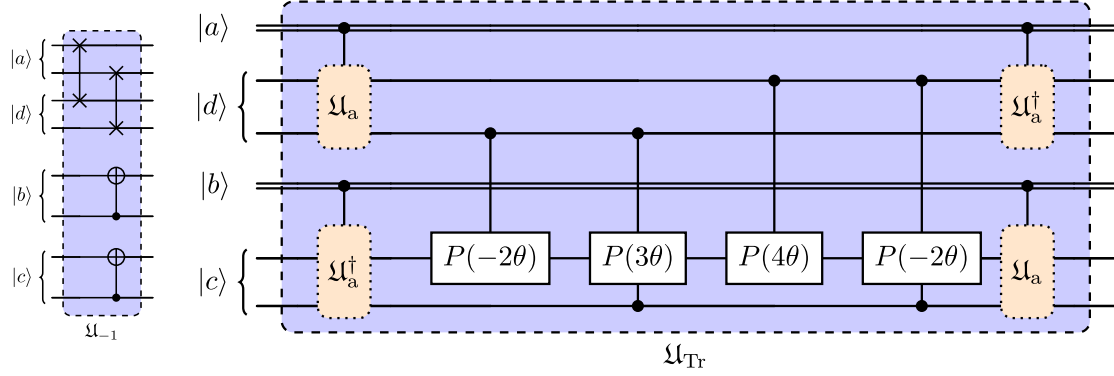FIG. 9. Quantum circuit implementing matrix element additions and productions in $\mathbf{F}_5$.

FIG. 10.   Inverse gate $\mathfrak{U}_{-1}$ and trace gate $\mathfrak{U}_{\mathrm{Tr}}$ for $\mathbb{BT}$ group with the two's component encoding.

TABLE V.   Addition.

	00	01	11
00	00	01	11
01	01	11	00
11	11	00	01

TABLE VI.   Multiplication.

	00	01	11
00	00	00	00
01	00	01	11
11	00	11	10

addition and multiplication table in Tables V and VI, the quantum circuit to implement matrix element additions $\mathfrak{U}_{\mathrm{a}}$ and productions $\mathfrak{U}_{\mathrm{p}}$ are shown in Fig. 9. The construction of the group multiplication circuit $\mathfrak{U}_{\times}$ is similar as Fig. 7 in the main text. The inverse and trace operations follow the same rules as Eqs. (11) and (12) in the main text, but with different quantum circuits given in Fig. 10.

[1] T. D. Cohen, H. Lamm, S. Lawrence, and Y. Yamauchi (NuQS Collaboration), Quantum algorithms for transport coefficients in gauge theories, Phys. Rev. D **104**, 094514 (2021).

[2] F. Turro, A. Ciavarella, and X. Yao, Classical and quantum computing of shear viscosity for $2+1D$ SU(2) gauge theory, Phys. Rev. D **109**, 114511 (2024).

[3] R. C. Farrell, M. Illa, and M. J. Savage, Steps toward quantum simulations of hadronization and energy-loss in dense matter, arXiv:2405.06620.

[4] H. Lamm, S. Lawrence, and Y. Yamauchi (NuQS Collaboration), Parton physics on a quantum computer, Phys. Rev. Res. **2**, 013272 (2020).

[5] M. Kreshchuk, W. M. Kirby, G. Goldstein, H. Beauchemin, and P. J. Love, Quantum simulation of quantum field theory in the light-front formulation, Phys. Rev. A **105**, 032418 (2022).

[6] M. Echevarria, I. Egusquiza, E. Rico, and G. Schnell, Quantum simulation of light-front parton correlators, Phys. Rev. D **104**, 014512 (2021).

[7] C. W. Bauer, M. Freytsis, and B. Nachman, Simulating collider physics on quantum computers using effective field theories, Phys. Rev. Lett. **127**, 212001 (2021).

[8] T. Li, X. Guo, W. K. Lai, X. Liu, E. Wang, H. Xing, D.-B. Zhang, and S.-L. Zhu (QuNu Collaboration), Partonic collinear structure by quantum computing, Phys. Rev. D **105**, L111502 (2022).

[9] T. V. Zache, D. González-Cuadra, and P. Zoller, Fermion-qudit quantum processors for simulating lattice gauge theories with matter, Quantum **7**, 1140 (2023).

[10] A. Polkovnikov and V. Gritsev, Breakdown of the adiabatic limit in low-dimensional gapless systems, Nat. Phys. **4**, 477 (2008).

[11] S. Baum, M. Carena, N. R. Shah, C. E. M. Wagner, and Y. Wang, Nucleation is more than critical: A case study of the electroweak phase transition in the NMSSM, J. High Energy Phys. 03 (2021) 055.

[12] A. Riera, C. Gogolin, and J. Eisert, Thermalization in nature and on a quantum computer, Phys. Rev. Lett. **108**, 080402 (2012).

[13] A. M. Czajka, Z.-B. Kang, H. Ma, and F. Zhao, Quantum simulation of chiral phase transitions, J. High Energy Phys. 08 (2022) 209.

[14] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

[15] S. P. Jordan, H. Krovi, K. S. Lee, and J. Preskill, BQP-completeness of scattering in scalar quantum field theory, Quantum **2**, 44 (2018).

[16] M. C. Bañuls *et al.*, Simulating lattice gauge theories within quantum technologies, Eur. Phys. J. D **74**, 165 (2020).

[17] C. W. Bauer *et al.*, Quantum simulation for high-energy physics, PRX Quantum **4**, 027001 (2023).

[18] A. Di Meglio *et al.*, Quantum computing for high-energy physics: State of the art and challenges. Summary of the QC4HEP working group, PRX Quantum **5**, 037001 (2024).

[19] R. Anishetty, M. Mathur, and I. Raychowdhury, Irreducible SU(3) Schwinger bosons, J. Math. Phys. (N.Y.) **50**, 053503 (2009).

[20] R. Anishetty and I. Raychowdhury, SU(2) lattice gauge theory: Local dynamics on nonintersecting electric flux loops, Phys. Rev. D **90**, 114503 (2014).

[21] I. Raychowdhury and J. R. Stryker, Loop, string, and hadron dynamics in SU(2) Hamiltonian lattice gauge theories, Phys. Rev. D **101**, 114502 (2020).

[22] E. Zohar, J. I. Cirac, and B. Reznik, Quantum simulations of gauge theories with ultracold atoms: Local gauge invariance from angular momentum conservation, Phys. Rev. A **88**, 023617 (2013).

[23] D. B. Kaplan and J. R. Stryker, Gauss's law, duality, and the Hamiltonian formulation of u(1) lattice gauge theory, Phys. Rev. D **102**, 094515 (2020).

[24] J. Bender and E. Zohar, Gauge redundancy-free formulation of compact QED with dynamical matter for quantum and classical computations, Phys. Rev. D **102**, 114517 (2020).

[25] A. Yamamoto, Real-time simulation of $(2 + 1)$-dimensional lattice gauge theory on qubits, Prog. Theor. Exp. Phys. **2021**, 013B06 (2021).

[26] C. W. Bauer and D. M. Grabowska, Efficient representation for simulating u(1) gauge theories on digital quantum computers at all values of the coupling, Phys. Rev. D **107**, L031503 (2023).

[27] A. N. Ciavarella and C. W. Bauer, Quantum simulation of SU(3) lattice Yang Mills theory at leading order in large n, arXiv:2402.10265.

[28] A. J. Buser, T. Bhattacharya, L. Cincio, and R. Gupta, State preparation and measurement in a quantum simulation of the $o(3)$ sigma model, Phys. Rev. D **102**, 114514 (2020).

[29] T. Bhattacharya, A. J. Buser, S. Chandrasekharan, R. Gupta, and H. Singh, Qubit regularization of asymptotic freedom, Phys. Rev. Lett. **126**, 172001 (2021).

[30] A. Alexandru, P. F. Bedaque, A. Carosso, and A. Sheng, Universality of a truncated sigma-model, Phys. Lett. B **832**, 137230 (2022).

[31] E. Zohar and M. Burrello, Formulation of lattice gauge theories for quantum simulations, Phys. Rev. D **91**, 054506 (2015).

[32] H. Lamm, S. Lawrence, and Y. Yamauchi (NuQS Collaboration), General methods for digital quantum simulation of gauge theories, Phys. Rev. D **100**, 034518 (2019).

[33] A. F. Shaw, P. Lougovski, J. R. Stryker, and N. Wiebe, Quantum algorithms for simulating the lattice Schwinger model, Quantum **4**, 306 (2020).

[34] Y. Tong, V. V. Albert, J. R. McClean, J. Preskill, and Y. Su, Provably accurate simulation of gauge theories and bosonic systems, Quantum **6**, 816 (2022).

[35] Y. Ji, H. Lamm, and S. Zhu (NuQS Collaboration), Gluon field digitization via group space decimation for quantum computers, Phys. Rev. D **102**, 114513 (2020).

[36] M. Carena, E. J. Gustafson, H. Lamm, Y.-Y. Li, and W. Liu, Gauge theory couplings on anisotropic lattices, Phys. Rev. D **106**, 114504 (2022).

[37] C. W. Bauer, I. D'Andrea, M. Freytsis, and D. M. Grabowska, A new basis for Hamiltonian SU(2) simulations, Phys. Rev. D **109**, 074501 (2024).

[38] A. N. Ciavarella, Quantum simulation of lattice QCD with improved Hamiltonians, Phys. Rev. D **108**, 094513 (2023).

[39] M. Hanada, J. Liu, E. Rinaldi, and M. Tezuka, Estimating truncation effects of quantum bosonic systems using sampling algorithms, Mach. Learn. Sci. Tech. **4**, 045021 (2023).

[40] M. Creutz, L. Jacobs, and C. Rebbi, Monte Carlo study of Abelian lattice gauge theories, Phys. Rev. D **20**, 1915 (1979).

[41] M. Creutz and M. Okawa, Generalized actions in $Z(p)$ lattice gauge theory, Nucl. Phys. **B220**, 149 (1983).

[42] G. Bhanot and C. Rebbi, Monte Carlo simulations of lattice models with finite subgroups of SU(3) as gauge groups, Phys. Rev. D **24**, 3319 (1981).

[43] D. Petcher and D. H. Weingarten, Monte Carlo calculations and a model of the phase structure for gauge theories on discrete subgroups of SU(2). Phys. Rev. D **22**, 2465 (1980).

[44] G. Bhanot, SU(3) lattice gauge theory in 4 dimensions with a modified Wilson action, Phys. Lett. **108B**, 337 (1982).

[45] E. J. Gustafson, H. Lamm, F. Lovelace, and D. Musk, Primitive quantum gates for an SU(2) discrete subgroup: Binary tetrahedral, Phys. Rev. D **106**, 114501 (2022).

[46] D. C. Hackett, K. Howe, C. Hughes, W. Jay, E. T. Neil, and J. N. Simone, Digitizing gauge fields: Lattice Monte Carlo results for future quantum computers, Phys. Rev. A **99**, 062341 (2019).

[47] M. S. Alam, S. Hadfield, H. Lamm, and A. C. Y. Li (SQMS Collaboration), Primitive quantum gates for dihedral gauge theories, Phys. Rev. D **105**, 114501 (2022).

[48] Y. Ji, H. Lamm, and S. Zhu (NuQS Collaboration), Gluon digitization via character expansion for quantum computers, Phys. Rev. D **107**, 114503 (2023).

[49] A. Alexandru, P. F. Bedaque, R. Brett, and H. Lamm, Spectrum of digitized QCD: Glueballs in a S(1080) gauge theory, Phys. Rev. D **105**, 114508 (2022).

[50] D. H. Weingarten and D. N. Petcher, Monte Carlo integration for lattice gauge theories with fermions, Phys. Lett. **99B**, 333 (1981).

[51] D. Weingarten, Monte Carlo evaluation of hadron masses in lattice gauge theories with fermions, Phys. Lett. **109B**, 57 (1982).

[52] A. Alexandru, P. F. Bedaque, S. Harmalkar, H. Lamm, S. Lawrence, and N. C. Warrington (NuQS Collaboration), Gluon field digitization for quantum computers, Phys. Rev. D **100**, 114501 (2019).

[53] E. J. Gustafson, H. Lamm, and F. Lovelace, Primitive quantum gates for an SU(2) discrete subgroup: Binary octahedral, Phys. Rev. D **109**, 054503 (2024).

[54] E. J. Gustafson and H. Lamm, Robustness of gauge digitization to quantum noise, arXiv:2301.10207.

[55] E. J. Gustafson, Y. Ji, H. Lamm, E. M. Murairi, and S. Zhu, Primitive quantum gates for an SU(3) discrete subgroup: $\Sigma(36 \times 3)$, arXiv:2405.05973.

[56] J. J. Vartiainen, M. Möttönen, and M. M. Salomaa, Efficient decomposition of quantum gates, Phys. Rev. Lett. **92,** 177902 (2004).

[57] N. P. D. Sawaya, T. Menke, T. H. Kyaw, S. Johri, A. Aspuru-Guzik, and G. G. Guerreschi, Resource-efficient digital quantum simulation of d-level systems for photonic, vibrational, and spin-s Hamiltonians, npj Quantum Inf. **6,** 49 (2020).

[58] O. Di Matteo, A. McCoy, P. Gysbers, T. Miyagi, R. M. Woloshyn, and P. Navrátil, Improving Hamiltonian encodings with the Gray code, Phys. Rev. A **103,** 042405 (2021).

[59] C. C. Chang, K. S. McElvain, E. Rrapaj, and Y. Wu, Improving Schrödinger equation implementations with Gray code for adiabatic quantum computers, PRX Quantum **3,** 020356 (2022).

[60] H. Lamm, S. Lawrence, and Y. Yamauchi (NuQS Collaboration), General methods for digital quantum simulation of gauge theories, Phys. Rev. D **100,** 034518 (2019).

[61] C. Moore, D. Rockmore, and A. Russell, Generic quantum Fourier transforms, arXiv:quant-ph/0304064.

[62] I. L. Chuang and M. A. Nielsen, Prescription for experimental determination of the dynamics of a quantum black box, J. Mod. Opt. **44,** 2455 (1997).

[63] A. Y. Kitaev, Quantum computations: Algorithms and error correction, Russ. Math. Surv. **52,** 1191 (1997).

[64] P. M. Q. Cruz and B. Murta, Shallow unitary decompositions of quantum Fredkin and Toffoli gates for connectivity-aware equivalent circuit averaging, APL Quantum **1,** 016105 (2024).

[65] T. Kim and B.-S. Choi, Efficient decomposition methods for controlled-Rn using a single ancillary qubit, Sci. Rep. **8,** 5445 (2018).

[66] G. Benenti, G. Casati, D. Rossini, and G. Strini, *Principles of Quantum Computation and Information (A Comprehensive Textbook)* (World Scientific, Singapore, 2019).

[67] A. Kan and Y. Nam, Lattice quantum chromodynamics and electrodynamics on a universal quantum computer, arXiv:2107.12769.

[68] M. Carena, H. Lamm, Y.-Y. Li, and W. Liu, Improved Hamiltonians for quantum simulations of gauge theories, Phys. Rev. Lett. **129,** 051601 (2022).

[69] Z. Davoudi, A. F. Shaw, and J. R. Stryker, General quantum algorithms for Hamiltonian simulation with applications to a non-Abelian lattice gauge theory, Quantum **7,** 1213 (2023).

[70] M. Rhodes, M. Kreshchuk, and S. Pathak, Exponential improvements in the simulation of lattice gauge theories using near-optimal techniques, arXiv:2405.10416.

[71] S. Lieu, R. Belyansky, J. T. Young, R. Lundgren, V. V. Albert, and A. V. Gorshkov, Symmetry breaking and error correction in open quantum systems, Phys. Rev. Lett. **125,** 240405 (2020).

[72] C. Bonati, A. Pelissetto, and E. Vicari, Breaking of gauge symmetry in lattice gauge theories, Phys. Rev. Lett. **127,** 091601 (2021).

[73] C. Bonati, A. Pelissetto, and E. Vicari, Lattice gauge theories in the presence of a linear gauge-symmetry breaking, Phys. Rev. E **104,** 014140 (2021).

[74] K. Stannigel, P. Hauke, D. Marcos, M. Hafezi, S. Diehl, M. Dalmonte, and P. Zoller, Constrained dynamics via the Zeno effect in quantum simulation: Implementing non-Abelian lattice gauge theories with cold atoms, Phys. Rev. Lett. **112,** 120406 (2014).

[75] M. C. Tran, Y. Su, D. Carney, and J. M. Taylor, Faster digital quantum simulation by symmetry protection, PRX Quantum **2,** 010323 (2021).

[76] H. Lamm, S. Lawrence, and Y. Yamauchi (NuQS Collaboration), Suppressing coherent gauge drift in quantum simulations, arXiv:2005.12688.

[77] J. R. Stryker, Oracles for Gauss's law on digital quantum computers, Phys. Rev. A **99,** 042301 (2019).

[78] A. Rajput, A. Roggero, and N. Wiebe, Quantum error correction with gauge symmetries, npj Quantum Inf. **9,** 41 (2023).

[79] E. Mathew and I. Raychowdhury, Protecting local and global symmetries in simulating $(1+1)$D non-Abelian gauge theories, Phys. Rev. D **106,** 054510 (2022).

[80] M. Van Damme, J. Mildenberger, F. Grusdt, P. Hauke, and J. C. Halimeh, Suppressing nonperturbative gauge errors in the thermodynamic limit using local pseudogenerators, arXiv:2110.08041.

[81] J. C. Halimeh and P. Hauke, Stabilizing gauge theories in quantum simulators: A brief review, arXiv:2204.13709.

[82] M. Carena, H. Lamm, Y.-Y. Li, and W. Liu, Quantum error thresholds for gauge-redundant digitizations of lattice field theories, arXiv:2402.16780.

[83] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O'Brien, Quantum error mitigation, Rev. Mod. Phys. **95,** 045005 (2023).

[84] E. Knill and R. Laflamme, Theory of quantum error-correcting codes, Phys. Rev. A **55,** 900 (1997).

[85] Quafu quantum computing cloud platform (2024), http://quafu.baqis.ac.cn.

[86] K. Yeter-Aydeniz, Z. Parks, A. Nair, E. Gustafson, A. F. Kemper, R. C. Pooser, Y. Meurice, and P. Dreher, Measuring NISQ gate-based qubit stability using a $1+1$ field theory and cycle benchmarking, Quantum Inf. Process. **22,** 96 (2023).

[87] R. C. Farrell, I. A. Chernyshev, S. J. M. Powell, N. A. Zemlevskiy, M. Illa, and M. J. Savage, Preparations for quantum simulations of quantum chromodynamics in $1+1$ dimensions: (I) axial gauge, Phys. Rev. D **107,** 054512 (2023).

[88] Y. Y. Atas, J. F. Haase, J. Zhang, V. Wei, S. M. L. Pfaendler, R. Lewis, and C. A. Muschik, Simulating one-dimensional quantum chromodynamics on a quantum computer: Real-time evolutions of tetra- and pentaquarks, Phys. Rev. Res. **5,** 033184 (2023).

[89] C. Charles, E. J. Gustafson, E. Hardt, F. Herren, N. Hogan, H. Lamm, S. Starecheski, R. S. Van de Water, and M. L. Wagman, Simulating $\mathbb{Z}_2$ lattice gauge theory on a quantum computer, Phys. Rev. E **109,** 015307 (2024).

[90] R. C. Farrell, M. Illa, A. N. Ciavarella, and M. J. Savage, Quantum simulations of hadron dynamics in the Schwinger model using 112 qubits, Phys. Rev. D **109,** 114510 (2024).

[91] E. Gustafson *et al.*, Large scale multi-node simulations of $\mathbb{Z}_2$ gauge theory quantum circuits using Google Cloud Platform, in *IEEE/ACM Second International Workshop on Quantum Computing Software* (2021), arXiv:2110.07482.

[92] E. Gustafson, Prospects for simulating a qudit based model of $(1 + 1)D$ scalar QED, Phys. Rev. D **103**, 114505 (2021).

[93] P. P. Popov, M. Meth, M. Lewenstein, P. Hauke, M. Ringbauer, E. Zohar, and V. Kasper, Variational quantum simulation of U(1) lattice gauge theories with qudit systems, Phys. Rev. Res. **6**, 013202 (2024).

[94] D. M. Kurkcuoglu, M. S. Alam, J. A. Job, A. C. Y. Li, A. Macridin, G. N. Perdue, and S. Providence, Quantum simulation of ϕ^4 theories in qudit systems, arXiv:2108.13357.

[95] G. Calajò, G. Magnifico, C. Edmunds, M. Ringbauer, S. Montangero, and P. Silvi, Digital quantum simulation of a

[96] D. González-Cuadra, T. V. Zache, J. Carrasco, B. Kraus, and P. Zoller, Hardware efficient quantum simulation of non-Abelian gauge theories with qudits on Rydberg platforms, Phys. Rev. Lett. **129**, 160501 (2022).

[97] M. Illa, C. E. P. Robin, and M. J. Savage, Qu8its for quantum simulations of lattice quantum chromodynamics, Phys. Rev. D **110**, 014507 (2024).

[98] T. Roy, T. Kim, A. Romanenko, and A. Grassellino, Qudit-based quantum computing with SRF cavities at Fermilab, Proc. Sci. LATTICE2023 (**2024**) 127.

$(1 + 1)$D SU(2) lattice gauge theory with ion qudits, arXiv:2402.07987.