A Compilation Framework for Block Encoding Classical Lie Algebra Matrices

Boyu Luan
Beijing 101 High School
13301017188@163.com

Yuejin Zhang
Beijing 101 High School
jason.zhang.1012028@gmail.com

Zihao Yan
Beijing 101 High School
jerryyan1215@gmail.com

August 2026

Abstract

We present a unified, constructive framework that compiles the defining representations of four classical Lie families into explicit block-encoding circuits. Block encodings are central to quantum algorithms, while Lie algebras govern symmetries and dynamics across a broad range of quantum systems. Our framework covers $\mathfrak{su}(n)$, $\mathfrak{so}_{2n+1}(\mathbb{C})$, $\mathfrak{sp}_{2n}(\mathbb{C})$, and $\mathfrak{so}_{2n}(\mathbb{C})$. Exploiting their defining identities, we decompose each matrix into diagonal and strictly upper-triangular primitives, encode the diagonal terms analytically, complete the triangular terms by a constructive Cholesky–Givens procedure, and assemble them through family-specific linear combinations of unitaries. In exact arithmetic, this yields exact block encodings with explicit normalizations and fully specified two-level circuits.

Let d be the representation dimension and $N = 2^{\lceil \log_2 d \rceil}$ its padded dimension. Under the strict joint-space two-level convention, the single-call count G and safe serial depth D are $G = D = d^2 - d + 14N$ for $\mathfrak{su}(n)$, $G = D = d^2 - d + 7N$ for $\mathfrak{so}_{2n+1}(\mathbb{C})$ and $\mathfrak{so}_{2n}(\mathbb{C})$, and $G = D = 2d^2 - 4d + 67N$ for $\mathfrak{sp}_{2n}(\mathbb{C})$. For $\mathfrak{su}(n)$, $\mathfrak{so}_{2n+1}(\mathbb{C})$, and $\mathfrak{so}_{2n}(\mathbb{C})$, our construction lowers the leading quadratic coefficient to 1, from 3/2 for direct isometry–Givens and 2 for dense Halmos dilation—asymptotic reductions of one third and one half, respectively. Cartan and fixed-width block-diagonal inputs further reduce the quadratic cost to linear scaling.

Numerical validation places all tested residuals at double-precision roundoff. Across 3380 physically motivated matrices from Rydberg, Majorana, and multimode bosonic models, the method beats both dense baselines in single-call resources for 1060 of 1480 instances from $\mathfrak{su}(n)$, $\mathfrak{so}_{2n+1}(\mathbb{C})$, and $\mathfrak{so}_{2n}(\mathbb{C})$, including all tested Rydberg cases with $d \geq 32$ and all even- and odd-dimensional Majorana cases with $d \geq 16$ and $d \geq 15$, respectively. At the largest Majorana dimensions, it also improves the normalization-weighted cost over Halmos dilation. In a passive, bus-dominated $\mathfrak{sp}_{2n}(\mathbb{C})$ regime, it outperforms both baselines in single-call resources at $d \in \{16, 32, 64\}$ and in weighted cost at $d \in \{32, 64\}$.

These results convert classical Lie-algebra structure into exact, explicit, and quantitatively smaller block-encoding circuits. The counts apply in the canonical basis and exclude a nontrivial basis change.

Keywords: quantum computing; block encoding; classical Lie algebras; quantum circuit synthesis; linear combination of unitaries; Cholesky–Givens decomposition; quantum resource estimation.

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖公示

Contents

1 Introduction **3**
1.1 Background . . . 3
1.2 Main Results . . . 3

2 Preliminaries **3**
2.1 Lie Algebras . . . 3
2.2 Representations of Lie Algebras . . . 5
2.2.1 Finite-Dimensional Representations . . . 5
2.2.2 Representations of the Classical Families . . . 5
2.2.3 Input and Output of the Compilation Framework . . . 7
2.3 Block Encoding . . . 7
2.3.1 Definition . . . 8
2.3.2 Action on a Quantum State . . . 9

3 Block Encoding of Lie Algebras **9**
3.1 Decompositions of the Representation of Lie Algebras . . . 9
3.1.1 Basic Triangular Decompositions . . . 9
3.1.2 Compact Real Form of Type A_{n-1} . . . 10
3.1.3 Types B_n and D_n . . . 10
3.1.4 Type C_n . . . 10
3.2 Quantum Circuits for Diagonal Matrices . . . 11
3.2.1 Purely Imaginary Diagonal Matrices . . . 11
3.2.2 Complex Diagonal Matrices . . . 13
3.3 Quantum Circuits for Strictly Upper-Triangular Matrices . . . 14
3.4 Quantum Circuits for $\mathfrak{su}(n)$ . . . 16
3.5 Quantum Circuits for $\mathfrak{so}_{2n}(\mathbb{C})$ and $\mathfrak{so}_{2n+1}(\mathbb{C})$ . . . 17
3.6 Quantum Circuits for $\mathfrak{sp}_{2n}(\mathbb{C})$ . . . 19

4 Some Low-Complexity Cases **21**
4.1 Cartan Subalgebra . . . 21
4.1.1 The Structure of Cartan Subalgebra . . . 22
4.1.2 Explicit Block Encodings . . . 22
4.2 Block-Diagonal Subalgebras . . . 25
4.2.1 Block Structure . . . 25
4.2.2 Direct-Sum Block Encoding and Two-Level Factor Count . . . 26
4.2.3 Repeated Blocks . . . 27

5 Complexity of the Lie-Algebra-Aware Construction **28**
5.1 Resource conventions . . . 28
5.2 Type A_{n-1} . . . 29
5.3 Types B_n and D_n . . . 30
5.4 Type C_n . . . 30
5.5 Overall scaling and native-gate upper bound . . . 31

6 Competing Methods and Normalization-Weighted Cost **31**

7 Physical-Model Experiments **32**
7.1 R1: Three-Dimensional Random-Position Rydberg Networks . . . 32
7.2 R2: Dense Quadratic Majorana/SYK_2 Networks . . . 33
7.3 R3: Odd-Dimensional Majorana Networks . . . 33
7.4 R4: Multimode Bosonic Quadratic Systems . . . 34

8 Scope of the Structured Specializations **35**

9 Conclusion and Discussion **35**

A Reproducibility of the Numerical Results **36**
A.1 Numerical conventions . . . 36
A.2 Campaign specification . . . 36
A.3 Correctness evidence . . . 37

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

1 Introduction

1.1 Background

Lie groups and Lie algebras provide a natural framework for describing continuous symmetries and dynamics in quantum systems. Once a representation ρ and an ordered basis $\mathcal{B}$ are fixed, an abstract element $x \in \mathfrak{g}$ is represented by a concrete matrix $[\rho(x)]_{\mathcal{B}}$. Classical Lie families, including unitary, orthogonal, and symplectic types, arise naturally in the study of quantum dynamics and dynamical Lie algebras [1]. However, these representation matrices are not generally unitary and therefore cannot be directly implemented as quantum circuits.

Block encoding addresses this problem by embedding a normalized matrix A/α into a larger unitary U_A. It is a standard input model for quantum singular value transformation, which supports polynomial transformations of singular values and applications such as Hamiltonian simulation, matrix inversion, and spectral filtering [2, 3]. Linear combinations of unitaries provide a complementary method for assembling several implementable matrix components through PREP, SELECT, and UNPREP operations [4].

An abstract unitary embedding is useful only when the corresponding unitary can be compiled into an explicit and efficient circuit. Recent work has developed such circuits for sparse matrices and matrices with arithmetic, repeated-value, or dictionary-based structures [5, 6, 7]. These results motivate a complementary family-based approach: instead of treating every matrix independently, we exploit the defining identities shared by the classical Lie families. This paper considers the defining representations of $\mathfrak{su}(n)$, $\mathfrak{so}_{2n+1}(\mathbb{C})$, $\mathfrak{sp}_{2n}(\mathbb{C})$, and $\mathfrak{so}_{2n}(\mathbb{C})$, and develops a systematic route from their algebraic matrix structure to block-encoding quantum circuits.

1.2 Main Results

The main contribution of this paper is a structure-aware framework for constructing block encodings of the defining representations of the four classical Lie families. Given a representation matrix X_{in} in an arbitrary orthonormal input basis, we first transform it to a canonical basis, construct the corresponding block-encoding unitary U_0, and then return to the input basis through

$$U_{\text{in}} = (I_{2^a} \otimes \widetilde{W}^\dagger) U_0 (I_{2^a} \otimes \widetilde{W}),$$

where $\widetilde{W}$ is the N-dimensional, family-dependent extension of W defined below. Consequently, the algebraic construction needs to be carried out only once in the canonical basis.

In the canonical basis, we construct explicit block encodings for the diagonal and strictly upper-triangular components. Diagonal matrices admit exact one-ancilla circuits based on address-controlled one-qubit gates. For an upper-triangular matrix T, a Cholesky–Givens construction gives a unitary completion of T/α_T. These two primitives are then assembled according to the defining identities of the classical Lie families. In particular, type A_{n-1} uses the three-term decomposition $X = D + T - T^\dagger$, types B_n and D_n use the two-term decomposition $X = T - T^T$, and type C_n is reduced to eleven diagonal or triangular terms. For the generic unpruned cases, the corresponding LCU normalization factors are

$$\Lambda_A = \alpha_D + 2\alpha_T, \qquad \Lambda_O = 2\alpha_T, \qquad \Lambda_C = \frac{1}{2}\sum_{r=1}^{11} \alpha_r,$$

with outer selector registers of two, one, and four qubits, respectively.

The framework also identifies important cases in which the generic LCU assembly can be avoided. Cartan-subalgebra elements require only a diagonal block encoding together with a fixed factor such as σ_z or $J_0 = i\sigma_y$, reducing the number of parameter-dependent factors from quadratic to linear in the representation dimension. For block-diagonal subalgebras, the local encodings can be combined with normalization $\alpha = \max_j \alpha_j$ rather than $\sum_j \alpha_j$. When the blocks are identical, the same local circuit is reused through $U_A = I_m \otimes U_0$.

2 Preliminaries

2.1 Lie Algebras

Definition 2.1 (Lie algebra)**.** *Let $\mathbb{F}$ be either $\mathbb{R}$ or $\mathbb{C}$. A Lie algebra over $\mathbb{F}$ is a vector space $\mathfrak{g}$ over $\mathbb{F}$ equipped with a bilinear map*

$$[\,\cdot\,,\,\cdot\,]\colon \mathfrak{g} \times \mathfrak{g} \longrightarrow \mathfrak{g},$$

2026 S.-T. Yau High School Science Award 收件于2026年底前 科学奖论文编号

called the Lie bracket*, satisfying*

$$[x, y] = -[y, x], \tag{1}$$
$$[x, [y, z]] + [y, [z, x]] + [z, [x, y]] = 0 \tag{2}$$

for all $x, y, z \in \mathfrak{g}$*. The second identity is called the Jacobi identity.*

A vector subspace $\mathfrak{h} \subseteq \mathfrak{g}$ is a Lie subalgebra if

$$[\mathfrak{h}, \mathfrak{h}] \subseteq \mathfrak{h}.$$

It is an ideal if

$$[\mathfrak{g}, \mathfrak{h}] \subseteq \mathfrak{h}.$$

A linear map

$$\varphi \colon \mathfrak{g} \longrightarrow \mathfrak{k}$$

between two Lie algebras is a Lie algebra homomorphism if

$$\varphi([x, y]) = [\varphi(x), \varphi(y)]$$

for all $x, y \in \mathfrak{g}$. If φ is bijective, it is called a Lie algebra isomorphism. Two isomorphic Lie algebras have the same abstract Lie algebraic structure even when they arise from different concrete realizations. A non-abelian Lie algebra $\mathfrak{g}$ is called simple if its only ideals are $\{0\}$ and $\mathfrak{g}$.

Let $\mathfrak{g}$ be a finite-dimensional complex simple Lie algebra and let $\mathfrak{h} \subseteq \mathfrak{g}$ be a Cartan subalgebra. For $\alpha \in \mathfrak{h}^*$, define the root space

$$\mathfrak{g}_\alpha = \{x \in \mathfrak{g} : [h, x] = \alpha(h)x \text{ for every } h \in \mathfrak{h}\},$$

The set of roots is

$$\Phi = \{\alpha \in \mathfrak{h}^* \setminus \{0\} : \mathfrak{g}_\alpha \neq \{0\}\},$$

and $\mathfrak{g}$ admits the root-space decomposition

$$\mathfrak{g} = \mathfrak{h} \oplus \bigoplus_{\alpha \in \Phi} \mathfrak{g}_\alpha.$$

Finite-dimensional complex simple Lie algebras are classified, up to isomorphism, by their irreducible root systems [11]. The four infinite classical families are

$$A_{n-1}, \qquad B_n, \qquad C_n, \qquad D_n.$$

The remaining five types

$$G_2, \qquad F_4, \qquad E_6, \qquad E_7, \qquad E_8$$

are exceptional and will not be considered here.

Let $\varepsilon_1, \ldots, \varepsilon_n$ be the standard orthonormal basis of $\mathbb{R}^n$. The root systems of the four classical families may be described as follows:

$$\Phi(A_{n-1}) = \{\varepsilon_i - \varepsilon_j : 1 \leq i \neq j \leq n\}, \tag{3}$$
$$\Phi(B_n) = \{\pm\varepsilon_i \pm \varepsilon_j : 1 \leq i < j \leq n\} \cup \{\pm\varepsilon_i : 1 \leq i \leq n\}, \tag{4}$$
$$\Phi(C_n) = \{\pm\varepsilon_i \pm \varepsilon_j : 1 \leq i < j \leq n\} \cup \{\pm 2\varepsilon_i : 1 \leq i \leq n\}, \tag{5}$$
$$\Phi(D_n) = \{\pm\varepsilon_i \pm \varepsilon_j : 1 \leq i < j \leq n\}. \tag{6}$$

In the expressions $\pm\varepsilon_i \pm \varepsilon_j$, the two signs are chosen independently. The root system of type A_{n-1} lies in the $(n-1)$-dimensional subspace

$$\{(x_1, \ldots, x_n) \in \mathbb{R}^n : x_1 + \cdots + x_n = 0\}.$$

The ranks and dimensions of the corresponding complex simple Lie algebras are

type	rank	dimension over $\mathbb{C}$
A_{n-1}	$n-1$	n^2-1
B_n	n	$n(2n+1)$
C_n	n	$n(2n+1)$
D_n	n	$n(2n-1)$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Let $\mathfrak{g}$ be a complex Lie algebra. A real Lie algebra $\mathfrak{g}_0$ is called a real form of $\mathfrak{g}$ if

$$\mathfrak{g}_0 \otimes_{\mathbb{R}} \mathbb{C} \cong \mathfrak{g}.$$

A real form is compact if it admits a positive-definite inner product $\langle \cdot, \cdot \rangle$ satisfying

$$\langle [x, y], z\rangle + \langle y, [x, z]\rangle = 0$$

for all $x, y, z \in \mathfrak{g}_0$. Every complex simple Lie algebra admits a compact real form, unique up to isomorphism.

In this paper, we work with the compact real form of A_{n-1}. For the types B_n, C_n, and D_n, we work with the complex simple Lie algebras.

2.2 Representations of Lie Algebras

2.2.1 Finite-Dimensional Representations

The space of all linear transformations on V is denoted by

$$\mathfrak{gl}(V) = \mathrm{End}_{\mathbb{F}}(V).$$

It becomes a Lie algebra under the commutator bracket

$$[A, B] = A \circ B - B \circ A.$$

Definition 2.2 (Finite-dimensional representation)**.** *A finite-dimensional representation of $\mathfrak{g}$ on V is a linear map*

$$\rho : \mathfrak{g} \longrightarrow \mathfrak{gl}(V)$$

satisfying

$$\rho([x, y]) = [\rho(x), \rho(y)] = \rho(x)\rho(y) - \rho(y)\rho(x)$$

for all $x, y \in \mathfrak{g}$.

Thus, an abstract Lie algebra element $x \in \mathfrak{g}$ is represented by a linear transformation

$$\rho(x) : V \longrightarrow V.$$

If

$$\dim_{\mathbb{F}} V = d,$$

then ρ is called a d-dimensional representation.

Let

$$\mathcal{B} = (v_1, \ldots, v_d)$$

be an ordered basis of V. After choosing this basis, each linear transformation $\rho(x)$ is represented by a matrix

$$[\rho(x)]_{\mathcal{B}} \in M_d(\mathbb{F}).$$

This gives the matrix representation

$$\rho_{\mathcal{B}} : \mathfrak{g} \longrightarrow M_d(\mathbb{F}), \qquad x \longmapsto [\rho(x)]_{\mathcal{B}}.$$

The representation property becomes

$$[\rho([x, y])]_{\mathcal{B}} = [\rho(x)]_{\mathcal{B}}[\rho(y)]_{\mathcal{B}} - [\rho(y)]_{\mathcal{B}}[\rho(x)]_{\mathcal{B}}.$$

2.2.2 Representations of the Classical Families

We now specify the representations used in this paper. Let $\mathfrak{g}_A^{\mathrm{cpt}}$ denote the compact real form of the complex simple Lie algebra of type A_{n-1}, and let $\mathfrak{g}_B$, $\mathfrak{g}_C$, and $\mathfrak{g}_D$ denote the complex simple Lie algebras of types B_n, C_n, and D_n.

For each family, we use its defining representation and fix a canonical basis of the corresponding representation space.

Type A_{n-1}. The defining representation of the compact real form of type A_{n-1} is

$$\rho_A : \mathfrak{g}_A^{\mathrm{cpt}} \longrightarrow \mathfrak{gl}_n(\mathbb{C}),$$

where $\mathfrak{gl}_n(\mathbb{C})$ is regarded as a real Lie algebra. In the canonical basis, its image is

$$\rho_A(\mathfrak{g}_A^{\mathrm{cpt}}) = \mathfrak{su}(n) = \left\{X \in M_n(\mathbb{C}) : X^\dagger = -X, \quad \mathrm{tr}(X) = 0\right\}.$$

2026 S. T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Type B_n. The defining representation of type B_n is

$$\rho_B : \mathfrak{g}_B \longrightarrow \mathfrak{gl}_{2n+1}(\mathbb{C}).$$

In the canonical orthogonal basis, its image is

$$\rho_B(\mathfrak{g}_B) = \mathfrak{so}_{2n+1}(\mathbb{C}) = \left\{X \in M_{2n+1}(\mathbb{C}) : X^T = -X\right\}.$$

Type C_n. Let

$$J_{2n} = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}.$$

The defining representation of type C_n is

$$\rho_C : \mathfrak{g}_C \longrightarrow \mathfrak{gl}_{2n}(\mathbb{C}).$$

In the canonical symplectic basis, its image is

$$\rho_C(\mathfrak{g}_C) = \mathfrak{sp}_{2n}(\mathbb{C}) = \left\{X \in M_{2n}(\mathbb{C}) : X^T J_{2n} + J_{2n} X = 0\right\}.$$

Type D_n. The defining representation of type D_n is

$$\rho_D : \mathfrak{g}_D \longrightarrow \mathfrak{gl}_{2n}(\mathbb{C}).$$

In the canonical orthogonal basis, its image is

$$\rho_D(\mathfrak{g}_D) = \mathfrak{so}_{2n}(\mathbb{C}) = \left\{X \in M_{2n}(\mathbb{C}) : X^T = -X\right\}.$$

Although a Lie algebra generally admits many inequivalent finite-dimensional representations, in this paper we restrict our attention to the defining representations described above.

Let

$$\mathcal{B}_0 = (e_1, \ldots, e_d)$$

be the canonical basis of the representation space V, and let

$$\mathcal{B}_{\text{in}} = (b_1, \ldots, b_d)$$

be the basis supplied as part of the input.

Throughout this paper, the input basis is assumed to be orthonormal with respect to the fixed Hermitian inner product on V.

Define the change-of-basis matrix

$$W = \left([b_1]_{\mathcal{B}_0} \quad [b_2]_{\mathcal{B}_0} \quad \cdots \quad [b_d]_{\mathcal{B}_0}\right).$$

Its j-th column is the coordinate vector of b_j in the canonical basis. Consequently, $[v]_{\mathcal{B}_0} = W[v]_{\mathcal{B}_{\text{in}}}$ for every $v \in V$.

Since both $\mathcal{B}_0$ and $\mathcal{B}_{\text{in}}$ are orthonormal, W is unitary: $W^{-1} = W^\dagger$.

Let

$$X_0 = [\rho(x)]_{\mathcal{B}_0} = W X_{\text{in}} W^\dagger$$

and

$$X_{\text{in}} = [\rho(x)]_{\mathcal{B}_{\text{in}}} = W^\dagger X_0 W.$$

Let $q = \lceil \log_2 d \rceil$ and $N = 2^q$. We use the same padded coordinate ordering for the input and canonical matrices. Let $P_\tau = I_N$ for $\tau \in \{A, B, D\}$, and let P_C be the fixed permutation from ordinary trailing padding to the paired symplectic ordering in Eq. (78). Define

$$\widetilde{X}_{\text{in}} = P_\tau (X_{\text{in}} \oplus 0_{N-d}) P_\tau^\dagger, \qquad \widetilde{X}_0 = P_\tau (X_0 \oplus 0_{N-d}) P_\tau^\dagger,$$

and

$$\widetilde{W} = P_\tau (W \oplus I_{N-d}) P_\tau^\dagger.$$

Then $\widetilde{W}$ is unitary and $\widetilde{X}_0 = \widetilde{W} \widetilde{X}_{\text{in}} \widetilde{W}^\dagger$. Thus $\widetilde{W} = W \oplus I_{N-d}$ for types A, B, and D, whereas type C also incorporates the paired symplectic padding permutation.

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Lemma 2.3 (Unitary change of basis)**.** *Suppose that U_0 is an (α, a, ϵ)-block encoding of the padded canonical representation matrix $\widetilde{X}_0$, so that*

$$\left\| \widetilde{X}_0 - \alpha \left(\langle 0|^{\otimes a} \otimes I_N \right) U_0 \left(|0\rangle^{\otimes a} \otimes I_N \right) \right\| \leq \epsilon.$$

Define

$$U_{\text{in}} = \left(I_{2^a} \otimes \widetilde{W}^\dagger \right) U_0 \left(I_{2^a} \otimes \widetilde{W} \right).$$

Then U_{in} is an (α, a, ϵ)-block encoding of $\widetilde{X}_{\text{in}}$.

Proof. Since $\widetilde{W}$ is unitary, U_{in} is also unitary. Moreover,

$$\begin{aligned} &\left(\langle 0|^{\otimes a} \otimes I_N \right) U_{\text{in}} \left(|0\rangle^{\otimes a} \otimes I_N \right) \\ &\quad = \widetilde{W}^\dagger \left[\left(\langle 0|^{\otimes a} \otimes I_N \right) U_0 \left(|0\rangle^{\otimes a} \otimes I_N \right) \right] \widetilde{W}. \end{aligned} \tag{7}$$

Therefore,

$$\begin{aligned} &\left\| \widetilde{X}_{\text{in}} - \alpha \left(\langle 0|^{\otimes a} \otimes I_N \right) U_{\text{in}} \left(|0\rangle^{\otimes a} \otimes I_N \right) \right\| \\ &\quad = \left\| \widetilde{W}^\dagger \left[\widetilde{X}_0 - \alpha \left(\langle 0|^{\otimes a} \otimes I_N \right) U_0 \left(|0\rangle^{\otimes a} \otimes I_N \right) \right] \widetilde{W} \right\| \\ &\quad \leq \epsilon. \end{aligned} \tag{8}$$

□

2.2.3 Input and Output of the Compilation Framework

The general mathematical compilation framework uses the explicit-matrix input

$$\mathcal{I} = (\tau, X_{\text{in}}, W),$$

where $\tau \in \{A, B, C, D\}$ specifies the Lie family, $X_{\text{in}} \in \mathbb{C}^{d \times d}$ is the matrix in the chosen orthonormal input basis, and W maps that basis to the canonical basis. The rank parameter is determined by τ and d, and the defining representation is fixed throughout this work; neither is therefore an independent input. For canonical-basis inputs, $W = I$.

The compiler verifies Lie-family membership and applies

$$\begin{aligned} X_{\text{in}} &\longmapsto X_0 = W X_{\text{in}} W^\dagger \\ &\longmapsto \text{canonical matrix decomposition} \\ &\longmapsto \text{family-dependent padding of the resulting primitives to dimension } N \\ &\longmapsto U_0 \\ &\longmapsto U_{\text{in}} = (I_{2^a} \otimes \widetilde{W}^\dagger) U_0 (I_{2^a} \otimes \widetilde{W}). \end{aligned}$$

The constructions in this work are exact, so ϵ is not a compiler input and $\epsilon = 0$ in exact arithmetic. The parameter ϵ is retained only in the general definition of block encoding and in the change-of-basis lemma. Floating-point residuals are numerical validation errors, while hardware-synthesis precision lies outside the present scope.

If the canonical-basis output is an N-dimensional state $|\phi\rangle_0$, then $|\phi\rangle_{\text{in}} = \widetilde{W}^\dagger |\phi\rangle_0$. For an operator Y_0, the corresponding input-basis result is $Y_{\text{in}} = \widetilde{W}^\dagger Y_0 \widetilde{W}$. Basis-independent scalar outputs require no final conversion. For a nontrivial basis transformation, the total circuit cost is

$$C_{\text{total}} = C_{\text{can}} + 2C(\widetilde{W}).$$

Consequently, for a fixed representation, it is sufficient to construct the quantum circuit in one canonical orthonormal basis. The circuit in any other orthonormal basis is obtained by conjugating the canonical circuit with the corresponding unitary change-of-basis transformation.

If the canonical circuit has gate complexity C_{can} and the padded change-of-basis unitary has gate complexity $C(\widetilde{W})$, then the total complexity is $C_{\text{total}} = C_{\text{can}} + 2C(\widetilde{W})$.

2.3 Block Encoding

Quantum circuits implement unitary transformations, whereas many operators arising in physics and mathematics are nonunitary. Block encoding embeds such an operator into a larger unitary matrix and provides a standard input model for quantum linear-algebra algorithms [2].

2026 Yau High School Science Award 仅用于2026丘成桐中学科学奖论文公示

2.3.1 Definition

Let

$$A \in \mathbb{C}^{d\times d}, \qquad q = \lceil \log_2 d \rceil, \qquad N = 2^q.$$

If $d < N$, then A is embedded into the N-dimensional system space by zero padding. For types A, B, and D, we use

$$\widetilde{A} = A \oplus 0_{N-d}.$$

Throughout this paper, d denotes the original matrix dimension and $N = 2^{\lceil \log_2 d \rceil}$ the padded system dimension. For type C, the zero padding is inserted in the paired symplectic ordering described in Eq. (78); this embedding is permutation-equivalent to ordinary trailing zero padding.

Definition 2.4 (Block encoding)**.** *A unitary U_A acting on $a+q$ qubits is an (α, a, ϵ)-block encoding of A if*

$$\left\| \widetilde{A} - \alpha(\langle 0^a| \otimes I_N) U_A (|0^a\rangle \otimes I_N) \right\|_2 \le \epsilon. \tag{9}$$

Here, $\alpha > 0$ is the normalization factor, a is the number of ancilla qubits, and $\epsilon \ge 0$ is the block-encoding error.

Equivalently, when the computational basis is ordered according to the ancilla register, U_A has the block form

$$U_A = \begin{pmatrix} \widetilde{A}/\alpha + E & * \\ * & * \end{pmatrix}, \qquad \|E\|_2 \le \frac{\epsilon}{\alpha}. \tag{10}$$

Only the upper-left block is required to approximate $\widetilde{A}/\alpha$. The remaining blocks are not uniquely determined; they complete the full matrix to a unitary operator.

When $\epsilon = 0$, the block encoding is exact:

$$(\langle 0^a| \otimes I_N) U_A (|0^a\rangle \otimes I_N) = \frac{\widetilde{A}}{\alpha}. \tag{11}$$

Since every projected block of a unitary is a contraction, an exact block encoding necessarily satisfies

$$\alpha \ge \|\widetilde{A}\|_2 = \|A\|_2. \tag{12}$$

The following standard LCU statement records the normalization and error bookkeeping used in all family-level assemblies [4, 2].

Lemma 2.5 (Linear combination of block encodings)**.** *Before forming an outer LCU, omit all zero coefficients and all algebraically zero primitives, and relabel the remaining K terms by $r = 0, \dots, K-1$. Such omitted terms contribute neither to K nor to Λ, PREP, or SELECT. If all primitives vanish, so that the complete input matrix is zero, no outer LCU is formed; instead, $X_c \otimes I_N$ is used as a standalone exact $(1,1,0)$-block encoding. Its normalization $\alpha = 1$ never contributes to an outer-LCU normalization. For the remaining terms, let U_r be an $(\alpha_r, a, \epsilon_r)$-block encoding of A_r, after padding idle ancillas if necessary. For*

$$A = \sum_{r=0}^{K-1} c_r A_r, \qquad c_r = |c_r| e^{i\phi_r}, \qquad \Lambda = \sum_{r=0}^{K-1} |c_r| \alpha_r > 0, \tag{13}$$

All terms use the same family-dependent embedding, which is linear, so $\widetilde{A} = \sum_r c_r \widetilde{A}_r$. Set $p_r = |c_r|\alpha_r/\Lambda$. Let $s = \lceil \log_2 K \rceil$ and choose $P|0^s\rangle = \sum_r \sqrt{p_r}|r\rangle$. Define

$$\mathrm{SELECT} = \sum_{r=0}^{K-1} |r\rangle\langle r| \otimes e^{i\phi_r} U_r + \sum_{r=K}^{2^s-1} |r\rangle\langle r| \otimes I \tag{14}$$

and

$$U_A = (P^\dagger \otimes I)\, \mathrm{SELECT}\, (P \otimes I). \tag{15}$$

Then U_A is a $(\Lambda, a+s, \sum_r |c_r|\epsilon_r)$-block encoding of A. In particular, it is exact when every constituent block encoding is exact.

Proof. Projection onto the selector and block-encoding signal states gives

$$\sum_{r=0}^{K-1} p_r e^{i\phi_r} \left(\frac{\widetilde{A}_r}{\alpha_r} + E_r \right) = \frac{\widetilde{A}}{\Lambda} + \sum_{r=0}^{K-1} p_r e^{i\phi_r} E_r, \tag{16}$$

where $\|E_r\|_2 \le \epsilon_r/\alpha_r$. The triangle inequality therefore bounds the unscaled projected-block error by $\Lambda^{-1}\sum_r |c_r|\epsilon_r$, proving the claim. □

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

2.3.2 Action on a Quantum State

Let $|\psi\rangle \in \mathbb{C}^N$ be a normalized system state. A physical d-dimensional state is embedded using the same family-dependent padded ordering as the matrix. In the exact case, applying U_A to the joint state $|0^a\rangle|\psi\rangle$ gives

$$U_A|0^a\rangle|\psi\rangle = |0^a\rangle\frac{\widetilde{A}|\psi\rangle}{\alpha} + |\Phi_\perp\rangle, \tag{17}$$

where

$$(\langle 0^a| \otimes I_N)|\Phi_\perp\rangle = 0.$$

Thus, conditioned on observing the ancilla register in $|0^a\rangle$, the system state is proportional to $\widetilde{A}|\psi\rangle$.

The corresponding success probability is

$$p_{\mathrm{succ}}(\psi) = \frac{\|\widetilde{A}|\psi\rangle\|_2^2}{\alpha^2}. \tag{18}$$

Therefore, α is not merely a mathematical scaling factor. A large value of α reduces the amplitude of the desired block and may increase the cost of postselection or amplitude amplification.

Block encoding serves as an input interface for quantum matrix algorithms. In particular, quantum singular value transformation can apply polynomial transformations to the singular values of a block-encoded matrix. For Hermitian matrices, this corresponds to polynomial transformations of their eigenvalues. This framework supports tasks such as Hamiltonian simulation, matrix inversion, and spectral filtering [2].

3 Block Encoding of Lie Algebras

3.1 Decompositions of the Representation of Lie Algebras

We decompose representation matrices of Lie algebras into diagonal and strictly upper triangular components. These decompositions provide the algebraic input for the block-encoding constructions developed in the following subsections.

All structural decompositions below are performed on the physical d-dimensional matrix X_0. For simplicity, we write

$$X = X_0$$

throughout this subsection. Before circuit construction, each resulting primitive is embedded into the common N-dimensional system space: types A, B, and D use trailing zero padding, whereas type C uses the paired symplectic ordering. We therefore decompose first and pad the primitives afterward; following the padding, the same symbols are reused for the N-dimensional primitives unless a tilde is displayed explicitly.

3.1.1 Basic Triangular Decompositions

For a matrix $M \in M_d(\mathbb{C})$, define its diagonal part and strictly upper triangular part by

$$D_M = \sum_{j=1}^{d} M_{jj}E_{jj}, \qquad T_M = \sum_{1\le j<k\le d} M_{jk}E_{jk},$$

respectively, where E_{jk} denotes the standard matrix unit.

The following elementary identities will be used repeatedly.

Lemma 3.1 (Basic triangular decompositions)**.** *If M is skew-Hermitian, then*

$$M^\dagger = -M \quad \Longrightarrow \quad M = D_M + T_M - T_M^\dagger.$$

If M is skew-symmetric, then

$$M^T = -M \quad \Longrightarrow \quad M = T_M - T_M^T.$$

Finally, if M is symmetric, then

$$M^T = M \quad \Longrightarrow \quad M = D_M + T_M + T_M^T.$$

Each decomposition is unique with respect to the fixed ordered standard basis.

2026 S.-T. Yau High School Science Award
仅用于2026年丘成桐中学科学奖论文公示

3.1.2 Compact Real Form of Type A_{n-1}

For the compact real form of type A_{n-1}, the defining representation identifies

$$\mathfrak{su}(n) = \left\{X \in M_n(\mathbb{C}) : X^\dagger = -X, \quad \mathrm{tr}(X) = 0\right\}.$$

Therefore, every $X \in \mathfrak{su}(n)$ admits the unique decomposition

$$X = D_X + T_X - T_X^\dagger.$$

Here D_X is diagonal and skew-Hermitian, while T_X is strictly upper triangular. More explicitly,

$$D_X = i\ \mathrm{diag}(d_1, \ldots, d_n), \qquad d_j \in \mathbb{R}, \qquad \sum_{j=1}^{n} d_j = 0.$$

3.1.3 Types B_n and D_n

In the fixed standard orthogonal basis, the defining representations of types B_n and D_n are

$$\mathfrak{so}_{2n+1}(\mathbb{C}) = \left\{X \in M_{2n+1}(\mathbb{C}) : X^T = -X\right\}$$

and

$$\mathfrak{so}_{2n}(\mathbb{C}) = \left\{X \in M_{2n}(\mathbb{C}) : X^T = -X\right\},$$

respectively.

A skew-symmetric matrix has zero diagonal. Hence, for either type,

$$X = T_X - T_X^T,$$

where T_X is the strictly upper triangular part of X. The local matrix decomposition is therefore the same for types B_n and D_n; only the matrix dimensions differ: $2n+1$ for type B_n, $2n$ for type D_n.

3.1.4 Type C_n

Let

$$J_{2n} = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}.$$

In its defining representation, the complex symplectic Lie algebra is

$$\mathfrak{sp}_{2n}(\mathbb{C}) = \left\{X \in M_{2n}(\mathbb{C}) : X^T J_{2n} + J_{2n} X = 0\right\}.$$

Writing X in $n \times n$ blocks, the symplectic condition is equivalent to

$$X = \begin{pmatrix} A & B \\ C & -A^T \end{pmatrix}, \qquad B^T = B, \qquad C^T = C.$$

Define

$$K = A - A^T, \qquad S = A + A^T, \qquad P = B + C, \qquad Q = B - C.$$

Then

$$K^T = -K, \qquad S^T = S, \qquad P^T = P, \qquad Q^T = Q.$$

Using the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

the symplectic matrix can be expressed as

$$X = \frac{1}{2}\left(I_2 \otimes K + \sigma_z \otimes S + \sigma_x \otimes P + i\sigma_y \otimes Q\right).$$

Indeed,

$$\frac{1}{2}\begin{pmatrix} K+S & P+Q \\ P-Q & K-S \end{pmatrix} = \begin{pmatrix} A & B \\ C & -A^T \end{pmatrix}.$$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Proposition 3.2 (Eleven-term decomposition for type C_n)**.** *Let T_K denote the strictly upper triangular part of K. For each $R \in \{S, P, Q\}$, let D_R and T_R denote the diagonal and strictly upper triangular parts of R, respectively. Then*

$$\begin{aligned} X = \frac{1}{2}\big(& I_2 \otimes T_K - I_2 \otimes T_K^T \\ & + \sigma_z \otimes D_S + \sigma_z \otimes T_S + \sigma_z \otimes T_S^T \\ & + \sigma_x \otimes D_P + \sigma_x \otimes T_P + \sigma_x \otimes T_P^T \\ & + i\sigma_y \otimes D_Q + i\sigma_y \otimes T_Q + i\sigma_y \otimes T_Q^T\big). \end{aligned}$$

Proof. Since K is skew-symmetric,

$$K = T_K - T_K^T.$$

Since S, P, and Q are symmetric,

$$R = D_R + T_R + T_R^T, \qquad R \in \{S, P, Q\}.$$

Substituting these decompositions into the preceding expression for X gives the stated formula. □

The decomposition expresses the matrix as a sum of diagonal and strictly upper-triangular terms, which are encoded below.

3.2 Quantum Circuits for Diagonal Matrices

3.2.1 Purely Imaginary Diagonal Matrices

Let

$$D = i\,\mathrm{diag}(d_0, d_1, \ldots, d_{n-1}), \qquad d_j \in \mathbb{R}, \quad \sum_{j=0}^{n-1} d_j = 0,$$

be the diagonal part of an element of $\mathfrak{su}(n)$.

For the N-dimensional circuit, define

$$\widetilde{d}_j = \begin{cases} d_j, & 0 \le j < n, \\ 0, & n \le j < N, \end{cases} \qquad \widetilde{D} = i\,\mathrm{diag}(\widetilde{d}_0, \ldots, \widetilde{d}_{N-1}) = D \oplus 0_{N-n}.$$

Assume $D \neq 0$ and choose

$$\alpha_D = \|D\|_2 = \|\widetilde{D}\|_2 = \max_{0 \le j < N} |\widetilde{d}_j|.$$

Define

$$\theta_j = \arcsin\left(\frac{\widetilde{d}_j}{\alpha_D}\right) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right],$$

and introduce the diagonal unitary

$$U_\theta = \mathrm{diag}\left(e^{i\theta_0}, e^{i\theta_1}, \ldots, e^{i\theta_{N-1}}\right).$$

For every j,

$$\frac{e^{i\theta_j} - e^{-i\theta_j}}{2} = i \sin\theta_j = \frac{i\widetilde{d}_j}{\alpha_D}.$$

Therefore,

$$\frac{\widetilde{D}}{\alpha_D} = \frac{U_\theta - U_\theta^\dagger}{2}. \tag{19}$$

Introduce one selector qubit c and define

$$\widetilde{\mathrm{SELECT}}_D = |0\rangle\langle 0|_c \otimes U_\theta + |1\rangle\langle 1|_c \otimes U_\theta^\dagger. \tag{20}$$

The relative minus sign required in Equation (19) is supplied by a Pauli Z gate on the selector:

$$\begin{aligned} \mathrm{SELECT}_D &= (Z_c \otimes I_N)\widetilde{\mathrm{SELECT}}_D \\ &= |0\rangle\langle 0|_c \otimes U_\theta - |1\rangle\langle 1|_c \otimes U_\theta^\dagger. \end{aligned} \tag{21}$$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Proposition 3.3 (Block encoding of purely imaginary diagonal matrices)**.** *The unitary*

$$B_D = (H_c \otimes I_N)\,\mathrm{SELECT}_D(H_c \otimes I_N) \tag{22}$$

is an exact $(\alpha_D, 1, 0)$*-block encoding of* D*. It satisfies*

$$(\langle 0|_c \otimes I_N) B_D (|0\rangle_c \otimes I_N) = \frac{U_\theta - U_\theta^\dagger}{2} = \frac{\widetilde{D}}{\alpha_D}. \tag{23}$$

Because

$$(\langle 0|_c \otimes I_N)(X_c \otimes I_N)(|0\rangle_c \otimes I_N) = 0,$$

if $D = 0$, it is omitted from any outer LCU. If the complete input matrix is zero and a standalone encoding is required, one may take $\alpha_D = 1$ and use

$$B_D = X_c \otimes I_N.$$

For distinct computational-basis labels p and q, and for $V \in U(2)$, let $G_{p,q}(V)$ denote the unitary that acts as V on the ordered subspace $\mathrm{span}\{|p\rangle, |q\rangle\}$ and as the identity on its orthogonal complement. Composite labels such as $p = (0, j)$ include both the ancilla and system addresses.

Use the convention

$$R_z(\varphi) = \begin{pmatrix} e^{-i\varphi/2} & 0 \\ 0 & e^{i\varphi/2} \end{pmatrix}.$$

For each $0 \le j < N$, define

$$\begin{aligned} V_j &= I_c \otimes (I_N - |j\rangle\langle j|) + R_z(-2\theta_j)_c \otimes |j\rangle\langle j| \\ &= G_{(0,j),(1,j)}\left(\begin{pmatrix} e^{i\theta_j} & 0 \\ 0 & e^{-i\theta_j} \end{pmatrix}\right), \end{aligned} \tag{24}$$

The factor V_j acts nontrivially only on

$$\mathrm{span}\left\{|0\rangle_c|j\rangle, |1\rangle_c|j\rangle\right\}.$$

and,

$$V_j|0\rangle_c|j\rangle = e^{i\theta_j}|0\rangle_c|j\rangle, \quad V_j|1\rangle_c|j\rangle = e^{-i\theta_j}|1\rangle_c|j\rangle.$$

On every basis state whose system address is not j, the factor V_j acts as the identity.

The factors act on mutually orthogonal address subspaces and therefore commute. Their product is

$$\prod_{j=0}^{N-1} V_j = |0\rangle\langle 0|_c \otimes U_\theta + |1\rangle\langle 1|_c \otimes U_\theta^\dagger = \widetilde{\mathrm{SELECT}}_D. \tag{25}$$

Consequently,

$$B_D = (H_c \otimes I_N)(Z_c \otimes I_N)\left(\prod_{j=0}^{N-1} V_j\right)(H_c \otimes I_N). \tag{26}$$

Write the system address as

$$j = (b_{q-1} b_{q-2} \cdots b_1 b_0)_2.$$

The two basis states on which V_j acts are

$$|0\rangle_c|b_{q-1}\cdots b_0\rangle, \qquad |1\rangle_c|b_{q-1}\cdots b_0\rangle.$$

They differ only in the selector qubit. Hence the selector qubit is the target of $R_z(-2\theta_j)$, while every system qubit is an address control. A filled control is used when $b_\ell = 1$, and an open control is used when $b_\ell = 0$. No Gray-code permutation is required.

For instance, for $N = 4$, the complete circuit is shown in Figure 1.

For general $N = 2^q$, the central address-controlled rotation is repeated for every $j = 0, \ldots, N-1$. The circuit therefore uses one selector qubit, two Hadamard gates, one Pauli Z gate, and one nontrivial address-controlled R_z rotation for each nonzero diagonal entry.

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

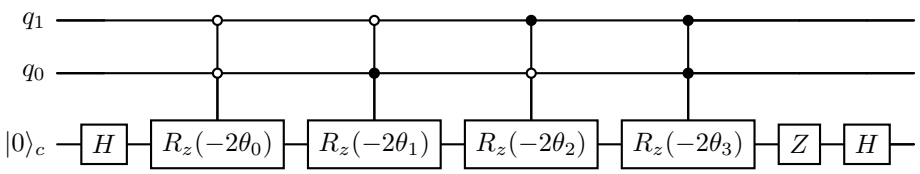


Figure 1: The imaginary diagonal block encoding for $N = 4$.

3.2.2 Complex Diagonal Matrices

We now consider the complex diagonal matrix. Write

$$D = \mathrm{diag}(z_0, z_1, \ldots, z_{N-1}) \in M_N(\mathbb{C}).$$

Assume that $D \neq 0$ and choose

$$\alpha_D = \|D\|_2 = \max_{0 \leq j < N} |z_j|.$$

Write

$$z_j = r_j e^{i\phi_j}, \qquad r_j = |z_j|,$$

where we take $\phi_j = 0$ when $z_j = 0$, and define

$$\vartheta_j = \arccos\left(\frac{r_j}{\alpha_D}\right) \in \left[0, \frac{\pi}{2}\right].$$

Introduce the two diagonal unitaries

$$U_\pm = \mathrm{diag}\left(e^{i(\phi_0 \pm \vartheta_0)}, e^{i(\phi_1 \pm \vartheta_1)}, \ldots, e^{i(\phi_{N-1} \pm \vartheta_{N-1})}\right).$$

For every j,

$$\frac{e^{i(\phi_j + \vartheta_j)} + e^{i(\phi_j - \vartheta_j)}}{2} = e^{i\phi_j} \cos\vartheta_j = \frac{z_j}{\alpha_D}.$$

Therefore,

$$\frac{D}{\alpha_D} = \frac{U_+ + U_-}{2}. \tag{27}$$

Using one selector qubit c, define

$$\mathrm{SELECT}_D = |0\rangle\langle 0|_c \otimes U_+ + |1\rangle\langle 1|_c \otimes U_-. \tag{28}$$

Proposition 3.4 (Block encoding of complex diagonal matrices)**.** *The unitary*

$$B_D = (H_c \otimes I_N)\,\mathrm{SELECT}_D(H_c \otimes I_N) \tag{29}$$

is an exact $(\alpha_D, 1, 0)$-block encoding of D. It satisfies

$$(\langle 0|_c \otimes I_N) B_D (|0\rangle_c \otimes I_N) = \frac{U_+ + U_-}{2} = \frac{D}{\alpha_D}.$$

The case $D = 0$ is handled by the same convention as in the preceding subsubsection.

The two-level decomposition can be obtained directly from the construction above. For each $0 \leq j < N$, define

$$\begin{aligned} V_j^{(\mathbb{C})} &= I_c \otimes (I_N - |j\rangle\langle j|) + \left(e^{i\phi_j} R_z(-2\vartheta_j)\right)_c \otimes |j\rangle\langle j| \\ &= G_{(0,j),(1,j)}\left(\begin{pmatrix} e^{i(\phi_j + \vartheta_j)} & 0 \\ 0 & e^{i(\phi_j - \vartheta_j)} \end{pmatrix}\right). \end{aligned} \tag{30}$$

Each factor acts nontrivially only on

$$\mathrm{span}\,\{|0\rangle_c|j\rangle, |1\rangle_c|j\rangle\}.$$

As in the preceding construction, these subspaces are mutually orthogonal, so the factors commute and

$$\prod_{j=0}^{N-1} V_j^{(\mathbb{C})} = \mathrm{SELECT}_D. \tag{31}$$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Consequently,

$$B_D = (H_c \otimes I_N) \left(\prod_{j=0}^{N-1} V_j^{(\mathbb{C})} \right) (H_c \otimes I_N).$$

For convenience, define the one-qubit diagonal unitary

$$C_j = e^{i\phi_j} R_z(-2\vartheta_j) = \begin{pmatrix} e^{i(\phi_j+\vartheta_j)} & 0 \\ 0 & e^{i(\phi_j-\vartheta_j)} \end{pmatrix}.$$

The address controls are the same as those in the preceding section. In particular, for $N = 4$, the complete circuit is shown in Figure 2.

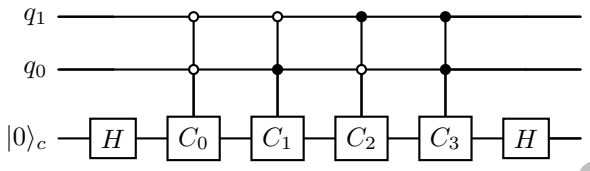


Figure 2: The block encoding of a complex diagonal matrix for $N = 4$.

The four controlled gates correspond respectively to the addresses

$$00, \qquad 01, \qquad 10, \qquad 11.$$

Thus, on the address state $|j\rangle$, the central part of the circuit acts on the selector qubit as

$$C_j|0\rangle_c = e^{i(\phi_j+\vartheta_j)}|0\rangle_c, \qquad C_j|1\rangle_c = e^{i(\phi_j-\vartheta_j)}|1\rangle_c.$$

Consequently, the complete central circuit implements

$$\prod_{j=0}^{N-1} V_j^{(\mathbb{C})} = |0\rangle\langle 0|_c \otimes U_+ + |1\rangle\langle 1|_c \otimes U_- = \mathrm{SELECT}_D\,.$$

For general $N = 2^q$, one address-controlled gate C_j is included for each $j = 0, \ldots, N-1$. The system qubits are filled or open controls according to the binary expansion of j, and the selector qubit is the target. The circuit therefore implements

$$B_D = (H_c \otimes I_N) \left(\prod_{j=0}^{N-1} V_j^{(\mathbb{C})} \right) (H_c \otimes I_N).$$

The phase factor $e^{i\phi_j}$ in C_j must not be discarded. Although it is a global phase of the local 2×2 matrix, after address control it becomes a relative phase between different system subspaces. Compared with the circuit for $M_n(i\mathbb{R})$, the controlled $R_z(-2\theta_j)$ gates are replaced by the controlled diagonal gates C_j, and the Pauli Z gate on the selector is no longer required.

3.3 Quantum Circuits for Strictly Upper-Triangular Matrices

Let $T \in \mathbb{C}^{N\times N}$ be strictly upper triangular. We construct an exact block encoding of T using one block ancilla and then give an explicit procedure for computing every two-level unitary in the resulting circuit.

If $T = 0$, the primitive is omitted from any outer LCU. If the complete input matrix is zero and a standalone encoding is required, $X_c \otimes I_N$ is an exact $(1, 1, 0)$-block encoding. Hence the construction below assumes $T \neq 0$.

Choose a normalization factor

$$\alpha_T > \|T\|_2 \tag{32}$$

and set

$$B = \frac{T}{\alpha_T}. \tag{33}$$

Then $\|B\|_2 < 1$, so $I_N - B^\dagger B$ is positive definite and admits the Cholesky factorization

$$I_N - B^\dagger B = LL^\dagger, \tag{34}$$

where L is lower triangular. Define the $2N \times N$ matrix

$$Q = \begin{pmatrix} B \\ L^\dagger \end{pmatrix}. \tag{35}$$

2026 Siyuan High School Science Award
仅用于2026思源高中科学奖论文公示

Equation (34) gives

$$Q^\dagger Q = B^\dagger B + LL^\dagger = I_N, \tag{36}$$

so the columns of Q are orthonormal. Moreover, both B and $L^\dagger$ are upper triangular. This common triangular structure permits the lower block of Q to be removed by a fixed column-by-column sequence of two-level Givens rotations.

We now specify this sequence constructively. Let $\widehat{Q}$ denote the current matrix, initialized as $\widehat{Q} = Q$. Process the columns in the order

$$j = 0, 1, \ldots, N-1,$$

and, within column j, process the lower-block rows in the order

$$i = j, j-1, \ldots, 0.$$

At the step indexed by (j, i), read the current pivot and target entry

$$x = (\widehat{Q})_{j,j}, \qquad y = (\widehat{Q})_{N+i,j}, \qquad \rho = \sqrt{|x|^2 + |y|^2}. \tag{37}$$

For $\rho \neq 0$, define

$$V(x, y) = \frac{1}{\rho} \begin{pmatrix} \overline{x} & \overline{y} \\ -y & x \end{pmatrix}. \tag{38}$$

If $\rho = 0$, set $V(x, y) = I_2$. The phase convention in Eq. (38) gives

$$V(x, y) \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \rho \\ 0 \end{pmatrix}, \qquad V(x, y)^\dagger V(x, y) = I_2, \qquad \det V(x, y) = 1. \tag{39}$$

Using the two-level notation introduced above, the full Givens unitary used at this step is

$$F_{j,i} = G_{(0,j),(1,i)}(V(x, y)). \tag{40}$$

It is computed from the current entries in Eq. (37), after which the entire matrix is updated by

$$\widehat{Q} \leftarrow F_{j,i} \widehat{Q}. \tag{41}$$

Thus the entry in position $(N+i, j)$ becomes zero. If this entry is already zero, the corresponding rotation may be omitted.

This elimination order does not recreate entries in previously processed columns. Indeed, before column j is processed, columns $0, \ldots, j-1$ have already been reduced. The upper-block row j is zero to the left of its diagonal, while the relevant entries of every lower-block row are either structurally zero or have already been eliminated. Hence both rows mixed by $F_{j,i}$ are zero in the completed columns.

Record the nontrivial rotations in their order of application as $F_1, F_2, \ldots, F_M$, and define

$$G = F_M F_{M-1} \cdots F_1. \tag{42}$$

Proposition 3.5 (Cholesky–Givens block encoding)**.** *The unitary*

$$U_T = G^\dagger = F_1^\dagger F_2^\dagger \cdots F_M^\dagger \tag{43}$$

is an exact $(\alpha_T, 1, 0)$-block encoding of T.

Proof. After all lower-block entries have been eliminated, one has

$$GQ = \begin{pmatrix} R \\ 0 \end{pmatrix}, \tag{44}$$

where R is upper triangular. Since G is unitary and $Q^\dagger Q = I_N$, Eq. (44) implies $R^\dagger R = I_N$. An upper-triangular unitary matrix is diagonal, and the convention in Eq. (38) makes every surviving diagonal pivot real and nonnegative. Consequently, $R = I_N$, and therefore

$$GQ = \begin{pmatrix} I_N \\ 0 \end{pmatrix}. \tag{45}$$

Since the upper $N \times N$ block of Q is T/α_T, this identity and $U_T = G^\dagger$ give the stated block encoding. □

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Although the factors in Eq. (43) are written in matrix-product order, a state entering the circuit encounters $F_M^\dagger$ first and $F_1^\dagger$ last. Every factor is explicitly known after the classical Cholesky factorization and the updates in Eq. (41); no unspecified unitary completion remains.

The active 2×2 matrix in Eq. (38) can be compiled into one-qubit rotations. Write

$$x = |x|e^{\mathrm{i}\phi_x}, \qquad y = |y|e^{\mathrm{i}\phi_y}.$$

One Euler-angle convention is

$$V(x, y) = R_z(\phi)R_y(\theta)R_z(\lambda), \tag{46}$$

where

$$\theta = 2\,\mathrm{atan2}(|y|, |x|), \qquad \phi = \phi_x + \phi_y - \pi, \qquad \lambda = \phi_x - \phi_y + \pi. \tag{47}$$

When $x = 0$ or $y = 0$, the phase of the zero entry may be chosen arbitrarily and absorbed into an adjacent R_z rotation.

The two basis labels acted on by $F_{j,i}$ are

$$|0\rangle_c|j\rangle_s \quad \text{and} \quad |1\rangle_c|i\rangle_s. \tag{48}$$

They differ in the block-ancilla bit and in $h_{j,i} = \mathrm{wt}(i \oplus j)$ system bits. The standard Gray-path construction therefore realizes $F_{j,i}$ using $2h_{j,i}$ multi-controlled X macros and one multi-controlled $V(x, y)$ macro. The latter is then replaced by the three rotations in Eq. (46). Controls on a zero-valued bit are implemented by conjugating the corresponding wire with X gates. This reduces the complete unitary U_T to the elementary gate model described in the preliminaries.

Corollary 3.6 (Resource cost of triangular block encoding)**.** *For column j, at most $j+1$ Givens rotations are required. Therefore,*

$$M \le \sum_{j=0}^{N-1}(j+1) = \frac{N(N+1)}{2}. \tag{49}$$

Thus the construction uses $O(N^2)$ two-level unitaries, $q = \log_2 N$ system qubits, and one block ancilla. With Gray-path compilation and linear-size multi-controlled-gate synthesis, the conservative elementary-gate cost is $O(N^2q^2)$. The classical preprocessing requires $O(N^3)$ arithmetic operations and $O(N^2)$ memory.

Further reductions require additional matrix structure, since a generic strictly upper-triangular matrix already contains $\Theta(N^2)$ independent entries.

3.4 Quantum Circuits for $\mathfrak{su}(n)$

The preceding decomposition gives

$$X = D + T - T^\dagger, \tag{50}$$

and

$$(\langle 0^a| \otimes I)\, U_D\, (|0^a\rangle \otimes I) = \frac{D}{\alpha_D}, \tag{51}$$

$$(\langle 0^a| \otimes I)\, U_T\, (|0^a\rangle \otimes I) = \frac{T}{\alpha_T}, \tag{52}$$

$$(\langle 0^a| \otimes I)\, U_T^\dagger\, (|0^a\rangle \otimes I) = \frac{T^\dagger}{\alpha_T}. \tag{53}$$

The explicit three-branch construction below describes the generic case $D \neq 0$ and $T \neq 0$. If either primitive vanishes, it is omitted according to Lemma 2.5, and K, Λ_A, PREP, and SELECT are formed only from the remaining branches. If both vanish, the standalone zero encoding is used. Define the three unitary branches

$$V_0 = U_D, \qquad V_1 = U_T, \qquad V_2 = -U_T^\dagger. \tag{54}$$

Let

$$\Lambda_A = \alpha_D + 2\alpha_T, \qquad \lambda_0 = \frac{\alpha_D}{\Lambda_A}, \qquad \lambda_1 = \lambda_2 = \frac{\alpha_T}{\Lambda_A}. \tag{55}$$

Two selector qubits are sufficient. We assign

$$|00\rangle \longleftrightarrow U_D, \qquad |01\rangle \longleftrightarrow U_T, \qquad |10\rangle \longleftrightarrow -U_T^\dagger,$$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖公示

while the unused state $|11\rangle$ selects the identity.

Let P_{prep} prepare

$$P_{\text{prep}}|00\rangle = |\chi_A\rangle = \sqrt{\lambda_0}|00\rangle + \sqrt{\lambda_1}|01\rangle + \sqrt{\lambda_2}|10\rangle. \tag{56}$$

Following the two-level construction of the three-term LCU, let $R_{02}(\theta_2)$ act on $\text{span}\{|00\rangle, |10\rangle\}$ and let $R_{01}(\theta_1)$ act on $\text{span}\{|00\rangle, |01\rangle\}$, with nontrivial block

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

Choose

$$\cos\theta_2 = \sqrt{\lambda_0 + \lambda_1}, \qquad \sin\theta_2 = \sqrt{\lambda_2}, \tag{57}$$

$$\cos\theta_1 = \sqrt{\frac{\lambda_0}{\lambda_0 + \lambda_1}}, \qquad \sin\theta_1 = \sqrt{\frac{\lambda_1}{\lambda_0 + \lambda_1}}. \tag{58}$$

Then

$$P_{\text{prep}} = R_{01}(\theta_1)R_{02}(\theta_2). \tag{59}$$

Define

$$\begin{aligned} \text{SELECT}_A = {} & |00\rangle\langle 00| \otimes U_D + |01\rangle\langle 01| \otimes U_T \\ & + |10\rangle\langle 10| \otimes (-U_T^\dagger) + |11\rangle\langle 11| \otimes I. \end{aligned} \tag{60}$$

Theorem 3.7 (Block encoding of $\mathfrak{su}(n)$)**.** *With the branches and weights defined above, the unitary*

$$U_X = (P_{\text{prep}}^\dagger \otimes I)\,\text{SELECT}_A(P_{\text{prep}} \otimes I) \tag{61}$$

is an exact $(\Lambda_A, a+2, 0)$-block encoding of X, where $\Lambda_A = \alpha_D + 2\alpha_T$.

Proof. Apply Lemma 2.5 to the three branches $(D, T, T^\dagger)$ with coefficients $(1, 1, -1)$ and normalizations $(\alpha_D, \alpha_T, \alpha_T)$. □

Here and below, $\mathcal{W}$ denotes all non-selector registers acted on by the selected branch unitaries, including the block-encoding ancillas and the padded system register, together with any family-specific factor register.

Figure 3 uses a convenient unitary extension of $R_{02}(\theta_2)$. The first uncontrolled R_y gate differs from the strict two-level rotation outside $\text{span}\{|00\rangle, |10\rangle\}$, but has the same action on the initial state $|00\rangle$. It therefore prepares the same LCU amplitudes and leaves the block encoding unchanged. The adjoint PREPARE circuit uses the reverse order with negative angles.

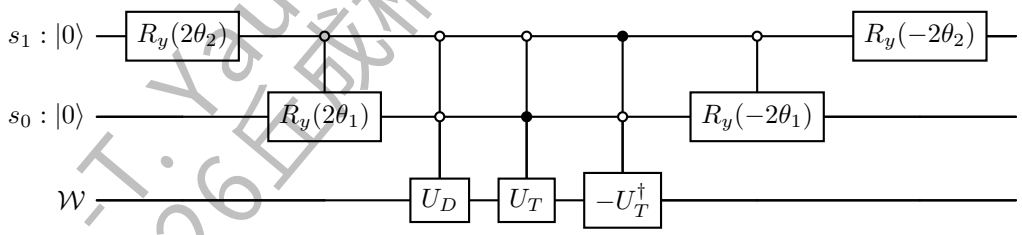


Figure 3: The block-encoding circuit for an element of $\mathfrak{su}(n)$.

The gate $-U_T^\dagger$ in Figure 3 means that the adjoint circuit is accompanied by a phase -1 on the selector word 10. Equivalently, one may implement the selector-only phase

$$\Phi_{10} = I_4 - 2|10\rangle\langle 10| \tag{62}$$

and use $U_T^\dagger$ in the corresponding controlled branch.

3.5 Quantum Circuits for $\mathfrak{so}_{2n}(\mathbb{C})$ and $\mathfrak{so}_{2n+1}(\mathbb{C})$

In the fixed standard orthonormal bases, $\mathfrak{so}_{2n}(\mathbb{C})$ and $\mathfrak{so}_{2n+1}(\mathbb{C})$ have the same skew-symmetric matrix form. We therefore treat them simultaneously by setting

$$d \in \{2n, 2n+1\} \tag{63}$$

2026 S.-T. Yau High School Science Award 仅用于2026丘成桐中学科学奖论文公示

and considering

$$\mathfrak{so}_d(\mathbb{C}) = \{X \in M_d(\mathbb{C}) : X^T = -X\}. \tag{64}$$

Let $X \in \mathfrak{so}_d(\mathbb{C})$ and let T be its strictly upper-triangular part. Since $X^T = -X$, every diagonal entry of X is zero and the strictly lower triangular part is $-T^T$. Therefore

$$X = T - T^T. \tag{65}$$

If $T = 0$, then $X = 0$ and the standalone zero encoding is used. The construction below therefore assumes $T \neq 0$. Their circuit constructions have the same form; only the dimension of the system register is different.

The preceding construction supplies an exact $(\alpha_T, a, 0)$-block encoding U_T of T, so that

$$\frac{T}{\alpha_T} = (\langle 0^a| \otimes I) U_T (|0^a\rangle \otimes I). \tag{66}$$

Lemma 3.8 (Transposition of a block encoding)**.** *The transpose U_T^T is an exact $(\alpha_T, a, 0)$-block encoding of T^T, satisfying*

$$(\langle 0^a| \otimes I) U_T^T (|0^a\rangle \otimes I) = \frac{T^T}{\alpha_T}. \tag{67}$$

Suppose that the known circuit is written as

$$U_T = F_m F_{m-1} \cdots F_1. \tag{68}$$

Then

$$U_T^T = F_1^T F_2^T \cdots F_m^T. \tag{69}$$

Thus the transpose circuit is obtained by reversing the chronological gate order and replacing each gate by its transpose.

Define the two unitary branches

$$V_0 = U_T, \qquad V_1 = -U_T^T. \tag{70}$$

Both branches have normalization factor α_T. Therefore the total normalization and the LCU probabilities are

$$\Lambda_O = 2\alpha_T, \qquad \lambda_0 = \lambda_1 = \frac{1}{2}. \tag{71}$$

Only one selector qubit is required, and its preparation unitary is the Hadamard gate:

$$P_{\text{prep}} = H, \qquad H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}. \tag{72}$$

Define

$$\text{SELECT}_O = |0\rangle\langle 0| \otimes U_T + |1\rangle\langle 1| \otimes (-U_T^T). \tag{73}$$

Theorem 3.9 (Block encoding of the orthogonal families)**.** *With the branches and weights defined above, the unitary*

$$U_X = (H \otimes I)\, \text{SELECT}_O (H \otimes I) \tag{74}$$

is an exact $(\Lambda_O, a+1, 0)$-block encoding of X, where $\Lambda_O = 2\alpha_T$.

Proof. Apply Lemma 2.5 to (T, T^T) with coefficients $(1, -1)$ and common normalization α_T. □

The relative minus sign in the second branch can be implemented by a Z gate on the selector qubit. Indeed,

$$\widetilde{\text{SELECT}}_O = |0\rangle\langle 0| \otimes U_T + |1\rangle\langle 1| \otimes U_T^T, \tag{75}$$

$$\text{SELECT}_O = (Z \otimes I)\widetilde{\text{SELECT}}_O. \tag{76}$$

Thus the signed SELECT operation can be drawn without using an abstract $-U_T^T$ gate, as shown in Figure 4.

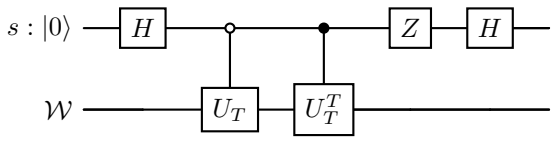


Figure 4: The block-encoding circuit for $\mathfrak{so}_{2n}(\mathbb{C})$ and $\mathfrak{so}_{2n+1}(\mathbb{C})$.

2026 S. T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

3.6 Quantum Circuits for $\mathfrak{sp}_{2n}(\mathbb{C})$

Writing X in $n \times n$ blocks, the symplectic condition is equivalent to

$$X = \begin{pmatrix} A & B \\ C & -A^T \end{pmatrix}, \qquad B^T = B, \qquad C^T = C. \tag{77}$$

The general zero-padding convention requires a slight modification for type C_n, because ordinary trailing zero padding does not necessarily preserve the canonical symplectic form. Suppose that the physical dimension is $d = 2n$ and that the padded dimension is $N = 2m \geq 2n$. Define

$$\widehat{A} = A \oplus 0_{m-n}, \qquad \widehat{B} = B \oplus 0_{m-n}, \qquad \widehat{C} = C \oplus 0_{m-n},$$

and use the structure-preserving padding

$$\widetilde{X} = \begin{pmatrix} \widehat{A} & \widehat{B} \\ \widehat{C} & -\widehat{A}^T \end{pmatrix}. \tag{78}$$

Equivalently, in terms of the fixed permutation defined above,

$$\widetilde{X} = P_C(X \oplus 0_{N-d})P_C^\dagger.$$

Because this embedding commutes with the linear combinations, transpose, and diagonal/triangular splitting used here, it is equivalent to applying the paired padding separately to the primitives of the physical decomposition. The padded matrix satisfies

$$\widetilde{X}^T J_{2m} + J_{2m}\widetilde{X} = 0.$$

For notational simplicity, after this structure-preserving padding, we henceforth relabel

$$\widetilde{X},\ \widehat{A},\ \widehat{B},\ \widehat{C}, \text{ and } m \quad \text{as} \quad X,\ A,\ B,\ C, \text{ and } n,$$

respectively.

Define

$$K = A - A^T, \qquad S = A + A^T, \qquad P = B + C, \qquad Q = B - C. \tag{79}$$

Then with the standard Pauli matrices, we have

$$X = \frac{1}{2}\left(I_2 \otimes K + \sigma_z \otimes S + \sigma_x \otimes P + i\sigma_y \otimes Q\right). \tag{80}$$

Let T_R and D_R denote the strictly upper triangular and diagonal parts of a matrix R. Since K is skew-symmetric and S, P, Q are symmetric,

$$K = T_K - T_K^T, \qquad R = T_R + D_R + T_R^T, \quad R \in \{S, P, Q\}. \tag{81}$$

Substitution into (80) gives the eleven-term decomposition

$$\begin{aligned} X = \frac{1}{2}\Big[& I_2 \otimes T_K - I_2 \otimes T_K^T \\ & + \sigma_z \otimes T_S + \sigma_z \otimes D_S + \sigma_z \otimes T_S^T \\ & + \sigma_x \otimes T_P + \sigma_x \otimes D_P + \sigma_x \otimes T_P^T \\ & + i\sigma_y \otimes T_Q + i\sigma_y \otimes D_Q + i\sigma_y \otimes T_Q^T \Big]. \end{aligned} \tag{82}$$

All transposes in this formula are ordinary matrix transposes, not adjoints. This distinction is necessary because the matrices are allowed to have complex entries.

Write

$$X = \frac{1}{2}\sum_{r=1}^{11} N_r, \qquad N_r = \tau_r \otimes M_r, \tag{83}$$

where the branch data are

r	τ_r	M_r	r	τ_r	M_r
1	I_2	T_K	2	$-I_2$	T_K^T
3	σ_z	T_S	4	σ_z	D_S
5	σ_z	T_S^T	6	σ_x	T_P
7	σ_x	D_P	8	σ_x	T_P^T
9	$i\sigma_y$	T_Q	10	$i\sigma_y$	D_Q
11	$i\sigma_y$	T_Q^T			

(84)

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖公示

The formulas below describe the generic unpruned case in which all eleven primitives are nonzero. If some M_r vanishes, it is omitted before Λ_C, PREP, and SELECT are defined, and the remaining $K_C \le 11$ branches are relabeled according to Lemma 2.5.

After padding with idle ancillas when necessary, assume that the preceding diagonal and strictly upper triangular constructions provide unitaries $U_1, \dots, U_{11}$ with a common number a of block-encoding ancillas. The preceding primitive constructions give

$$B_r = (\langle 0^a| \otimes I_n) U_r (|0^a\rangle \otimes I_n) = \frac{M_r}{\alpha_r}, \qquad r = 1, \dots, 11, \tag{85}$$

where $\alpha_r > 0$ is the normalization factor of the local block encoding of M_r. For each transposed triangular block, its circuit may be obtained from the untransposed circuit. If

$$U_T = F_m F_{m-1} \cdots F_1,$$

where the rightmost factor acts first, then

$$U_T^T = F_1^T F_2^T \cdots F_m^T \tag{86}$$

is a block encoding of T^T with the same normalization and error. On the structural qubit together with the common workspace, define

$$V_r = \tau_r \otimes U_r, \qquad r = 1, \dots, 11. \tag{87}$$

Because every τ_r is unitary, V_r is unitary and

$$(I_2 \otimes \langle 0^a| \otimes I_n) V_r (I_2 \otimes |0^a\rangle \otimes I_n) = \tau_r \otimes B_r. \tag{88}$$

Define

$$\beta_r = \frac{\alpha_r}{2}, \qquad \Lambda_C = \sum_{r=1}^{11} \beta_r = \frac{1}{2} \sum_{r=1}^{11} \alpha_r, \qquad p_r = \frac{\beta_r}{\Lambda_C} = \frac{\alpha_r}{\sum_{j=1}^{11} \alpha_j}. \tag{89}$$

Four selector qubits provide sixteen computational basis states. Assign

$$|0000\rangle \longleftrightarrow V_1, \quad |0001\rangle \longleftrightarrow V_2, \quad \dots \quad , |1010\rangle \longleftrightarrow V_{11},$$

and let P_{prep} prepare

$$P_{\mathrm{prep}} |0000\rangle = |\chi_C\rangle = \sum_{r=1}^{11} \sqrt{p_r}\, |r-1\rangle. \tag{90}$$

Let

$$q_k = p_{k+1}, \qquad s_k = \sum_{j=0}^{k} q_j, \qquad k = 0, \dots, 10. \tag{91}$$

For $k = 1, \dots, 10$, let $R_{0k}(\theta_k)$ be the two-level rotation on $\mathrm{span}\{|0\rangle, |k\rangle\}$ with nontrivial block

$$R(\theta_k) = \begin{pmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{pmatrix},$$

where

$$\cos\theta_k = \sqrt{\frac{s_{k-1}}{s_k}}, \qquad \sin\theta_k = \sqrt{\frac{q_k}{s_k}}. \tag{92}$$

Then

$$P_{\mathrm{prep}} = R_{01}(\theta_1) R_{02}(\theta_2) \cdots R_{0,10}(\theta_{10}), \tag{93}$$

where the rightmost factor acts first. In the generic eleven-branch case, PREP contains ten selector two-level rotations and UNPREP contains their ten adjoints. After pruning, the same construction uses $\lceil \log_2 K_C \rceil$ selector qubits and $K_C - 1$ rotations in each of PREP and UNPREP.

Define

$$\mathrm{SELECT}_C = \sum_{r=1}^{11} |r-1\rangle\langle r-1| \otimes V_r + \sum_{s=11}^{15} |s\rangle\langle s| \otimes I_{\mathcal{W}}, \tag{94}$$

where the five unused selector words act as the identity.

Theorem 3.10 (Block encoding of $\mathfrak{sp}_{2n}(\mathbb{C})$)**.** *With the branches and weights defined above, the unitary*

$$U_X = (P_{\mathrm{prep}}^\dagger \otimes I_{\mathcal{W}})\, \mathrm{SELECT}_C\, (P_{\mathrm{prep}} \otimes I_{\mathcal{W}}) \tag{95}$$

is an exact $(\Lambda_C, a+4, 0)$*-block encoding of* X*, where* $\Lambda_C = \frac{1}{2} \sum_{r=1}^{11} \alpha_r$*.*

2026 S. -T. Yau High School Science Award 仅用于2026丘成桐中学科学奖论文公示

Proof. Apply Lemma 2.5 to the eleven branches $\tau_r \otimes M_r$, each with coefficient $1/2$. Their primitive normalizations give $\Lambda_C = \frac{1}{2}\sum_r \alpha_r$. □

At register level, the complete circuit has the following form.

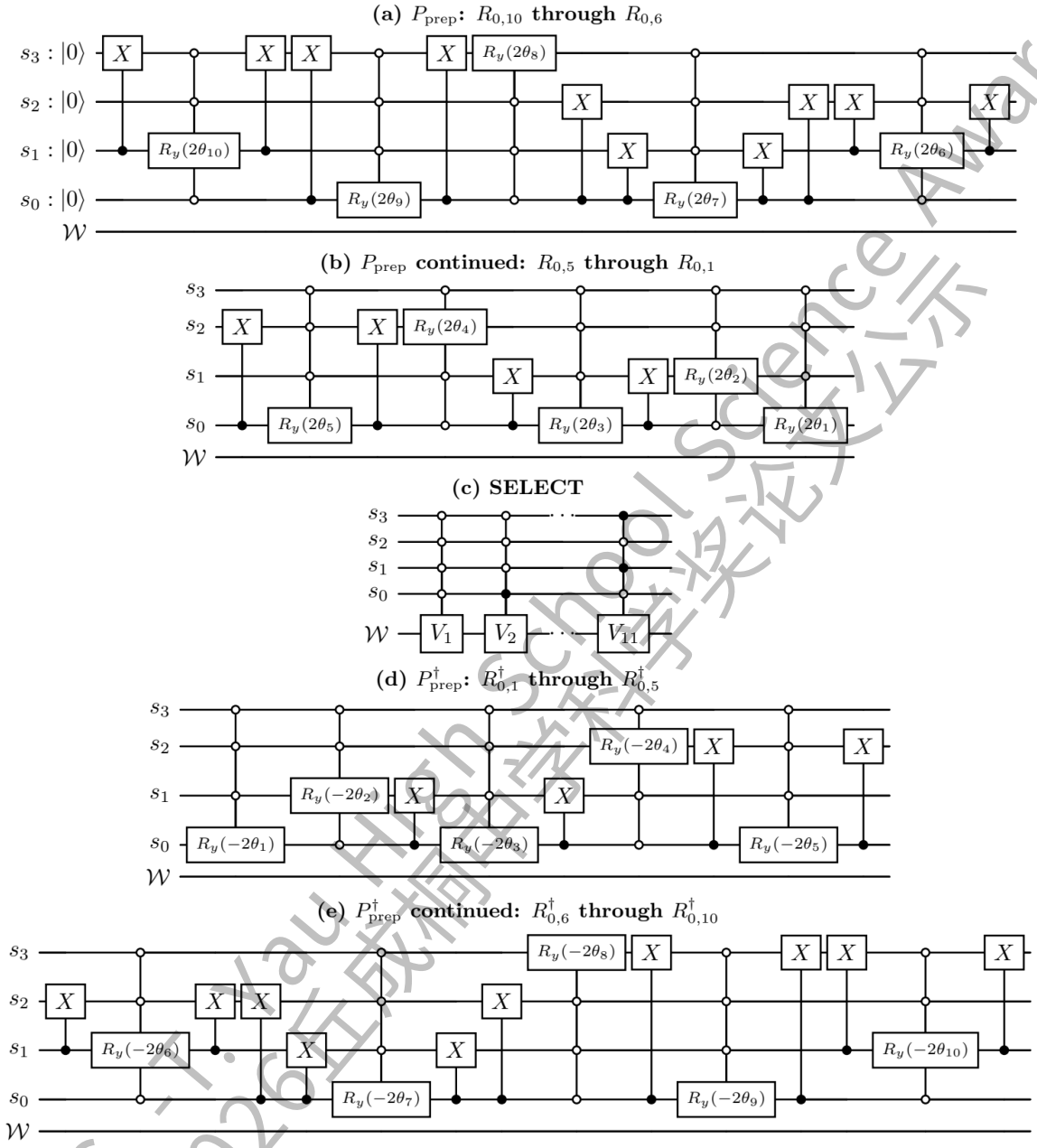


Figure 5: The complete LCU circuit for the block encoding of $X \in \mathfrak{sp}_{2n}(\mathbb{C})$.

4 Some Low-Complexity Cases

The preceding sections developed a general block-encoding framework for the four classical Lie families. We now examine several structured cases in which the representation matrices admit simpler decompositions and therefore require fewer circuit components.

4.1 Cartan Subalgebra

In the defining representations used above, the standard Cartan subalgebras are represented either by diagonal matrices or by direct sums of 2×2 skew-symmetric blocks [11].

2026 S. T. Yau High School Science Award
拟用于2026丘成桐中学科学奖论文公示

4.1.1 The Structure of Cartan Subalgebra

Type A_{n-1}. For the compact real form

$$\mathfrak{su}(n) = \left\{X \in M_n(\mathbb{C}) : X^\dagger = -X,\ \mathrm{tr}(X) = 0\right\},$$

a standard Cartan subalgebra is

$$\mathfrak{h}_A = \left\{ i\,\mathrm{diag}(t_1, \ldots, t_n) : t_j \in \mathbb{R},\ \sum_{j=1}^{n} t_j = 0 \right\}. \tag{96}$$

Thus every $H_A \in \mathfrak{h}_A$ has the form

$$H_A = i\,\mathrm{diag}(t_1, \ldots, t_n).$$

This matrix is already purely imaginary and diagonal, so its block encoding is given directly by the construction for $M_n(i\mathbb{R})$.

Types B_n and D_n. Define

$$J_0 = i\sigma_y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \qquad J(z) = zJ_0 = \begin{pmatrix} 0 & z \\ -z & 0 \end{pmatrix}.$$

In the standard orthogonal basis used in this paper, a Cartan subalgebra of $\mathfrak{so}_{2n+1}(\mathbb{C})$ is

$$\mathfrak{h}_B = \{J(z_1) \oplus \cdots \oplus J(z_n) \oplus 0 : z_j \in \mathbb{C}\}, \tag{97}$$

whereas a Cartan subalgebra of $\mathfrak{so}_{2n}(\mathbb{C})$ is

$$\mathfrak{h}_D = \{J(z_1) \oplus \cdots \oplus J(z_n) : z_j \in \mathbb{C}\}. \tag{98}$$

Let

$$Z = \mathrm{diag}(z_1, \ldots, z_n).$$

Then the two matrix forms can be written compactly as

$$H_B = (Z \otimes J_0) \oplus 0, \qquad H_D = Z \otimes J_0. \tag{99}$$

Type C_n. Recall that

$$\mathfrak{sp}_{2n}(\mathbb{C}) = \left\{X \in M_{2n}(\mathbb{C}) : X^T J_{2n} + J_{2n} X = 0\right\}, \qquad J_{2n} = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}.$$

In the canonical symplectic basis, a standard Cartan subalgebra is

$$\mathfrak{h}_C = \left\{ \begin{pmatrix} Z & 0 \\ 0 & -Z \end{pmatrix} : Z = \mathrm{diag}(z_1, \ldots, z_n),\ z_j \in \mathbb{C} \right\}. \tag{100}$$

Equivalently,

$$H_C = \mathrm{diag}(z_1, \ldots, z_n, -z_1, \ldots, -z_n) = \sigma_z \otimes Z. \tag{101}$$

4.1.2 Explicit Block Encodings

We now give the block-encoding unitaries for the four Cartan families. The internal circuit for a diagonal matrix is in Section 3.2. If a Cartan element is zero, we use the zero-matrix convention from Section 3.2: we set the normalization factor to 1 and take $X_c \otimes I_N$ as its block encoding. Hence the formulas below are stated for nonzero Cartan elements.

Type A_{n-1}. Set

$$q = \lceil \log_2 n \rceil, \qquad N = 2^q, \qquad \alpha_A = \|H_A\|_2 = \max_{1 \le k \le n} |t_k|.$$

For the zero-based computational addresses $j = 0, \ldots, N-1$, define

$$\widetilde{t}_j = \begin{cases} t_{j+1}, & 0 \le j < n, \\ 0, & n \le j < N, \end{cases} \qquad \theta_j = \arcsin\left(\frac{\widetilde{t}_j}{\alpha_A}\right).$$

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Let

$$\mathcal{M}_A = \sum_{j=0}^{N-1} R_z(-2\theta_j)_c \otimes |j\rangle\langle j| \tag{102}$$

be the address-controlled phase circuit. Thus, when the address register is in the state $|j\rangle$, $\mathcal{M}_A$ applies $R_z(-2\theta_j)$ to the block-encoding ancilla c. For every padding address $n \le j < N$, we have $\theta_j = 0$, so the corresponding operation is $R_z(0) = I$.

Since the padded matrix is

$$\widetilde{H}_A = H_A \oplus 0_{N-n} = i\,\mathrm{diag}(\widetilde{t}_0, \ldots, \widetilde{t}_{N-1}),$$

the purely imaginary diagonal construction gives

$$B_{H_A} = (H_c \otimes I_N)(Z_c \otimes I_N)\mathcal{M}_A(H_c \otimes I_N). \tag{103}$$

At the circuit level, the gates are applied from left to right as shown in Figure 6.

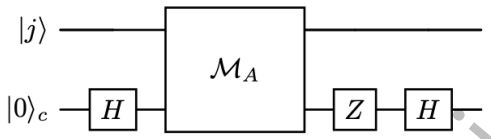


Figure 6: Block-encoding circuit for a type-A_{n-1} Cartan element.

For each address $j = 0, \ldots, N-1$,

$$\langle 0|HZR_z(-2\theta_j)H|0\rangle = i\sin\theta_j = \frac{i\widetilde{t}_j}{\alpha_A}.$$

Therefore,

$$(\langle 0|_c \otimes I_N)B_{H_A}(|0\rangle_c \otimes I_N) = \frac{\widetilde{H}_A}{\alpha_A},$$

and B_{H_A} is an exact $(\alpha_A, 1, 0)$ block encoding of the padded Cartan element $\widetilde{H}_A$.

Type B_n. Let

$$q = \lceil \log_2(2n+1) \rceil, \qquad N = 2^q, \qquad m = \frac{N}{2}.$$

Extend

$$Z = \mathrm{diag}(z_1, \ldots, z_n)$$

to the $m \times m$ diagonal matrix

$$\widetilde{Z} = \mathrm{diag}(z_1, \ldots, z_n, \underbrace{0, \ldots, 0}_{m-n}), \qquad \alpha_Z = \|\widetilde{Z}\|_2 = \max_{1 \le k \le n} |z_k|.$$

Identify the padded system space as

$$\mathbb{C}^N \cong \mathbb{C}^m \otimes \mathbb{C}^2, \qquad |j\rangle_j |r\rangle_r \longleftrightarrow |2j + r\rangle,$$

where $j = 0, \ldots, m-1$ labels the 2×2 blocks and $r \in \{0, 1\}$ labels the two coordinates within each block. The blocks $j = 0, \ldots, n-1$ represent the physical pairs, the basis state $|n\rangle_j |0\rangle_r$ represents the final physical zero coordinate, and all remaining basis states are padding coordinates.

With this ordering, the padded type-B_n Cartan element is

$$\widetilde{H}_B = H_B \oplus 0_{N-(2n+1)} = \widetilde{Z} \otimes J_0. \tag{104}$$

Let $B_{\widetilde{Z}}$ be the diagonal block encoding of $\widetilde{Z}$, satisfying

$$(\langle 0|_c \otimes I_m)B_{\widetilde{Z}}(|0\rangle_c \otimes I_m) = \frac{\widetilde{Z}}{\alpha_Z}.$$

Define

$$B_{H_B} = B_{\widetilde{Z},(c,j)} \otimes (J_0)_r. \tag{105}$$

2025 S.-T. Yau High School Science Award
仅用于2026江成梅中学科学论文公示

Thus c is the only block-encoding ancilla, while j and r are both parts of the system register. No additional flag qubit is required.

At the structured-block level, the circuit is shown below. Since $J_0 = ZX$, the r register is acted on first by X and then by Z.

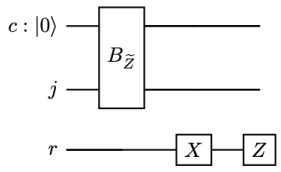


Figure 7: Block-encoding circuit for a type-B_n Cartan element.

Therefore,

$$(\langle 0|_c \otimes I_N) B_{H_B} (|0\rangle_c \otimes I_N) = \frac{\widetilde{Z}}{\alpha_Z} \otimes J_0 = \frac{\widetilde{H}_B}{\alpha_Z}.$$

Thus B_{H_B} is an exact $(\alpha_Z, 1, 0)$ block encoding of the padded Cartan element $\widetilde{H}_B$.

Type C_n. For types C_n and D_n, set

$$q = \lceil \log_2(2n) \rceil, \qquad N = 2^q, \qquad m = \frac{N}{2}, \qquad \widetilde{Z} = Z \oplus 0_{m-n} = \mathrm{diag}(\widetilde{z}_0, \ldots, \widetilde{z}_{m-1}),$$

where $\widetilde{z}_j = z_{j+1}$ for $0 \le j < n$ and $\widetilde{z}_j = 0$ otherwise. Let

$$\alpha_Z = \max_{0 \le j < m} |\widetilde{z}_j|, \qquad \phi_j = \begin{cases} \arg(\widetilde{z}_j), & \widetilde{z}_j \neq 0, \\ 0, & \widetilde{z}_j = 0, \end{cases} \qquad \vartheta_j = \arccos\left(\frac{|\widetilde{z}_j|}{\alpha_Z}\right),$$

and define

$$\mathcal{M}_{\widetilde{Z}} = \sum_{j=0}^{m-1} \left(e^{i\phi_j} R_z(-2\vartheta_j)\right)_c \otimes |j\rangle\langle j|, \qquad B_{\widetilde{Z}} = (H_c \otimes I_m) \mathcal{M}_{\widetilde{Z}} (H_c \otimes I_m).$$

Let s be the structural qubit distinguishing the two m-dimensional blocks of the paired padding

$$\widetilde{H}_C = P_C (H_C \oplus 0_{N-2n}) P_C^\dagger = \sigma_z \otimes \widetilde{Z}.$$

Define

$$B_{H_C} = (\sigma_z)_s B_{\widetilde{Z},(c,j)}. \tag{106}$$

Thus, the circuit applies a single Pauli-Z gate to s and applies $B_{\widetilde{Z}}$ to the ancilla-address registers (c, j), as shown in Figure 8.

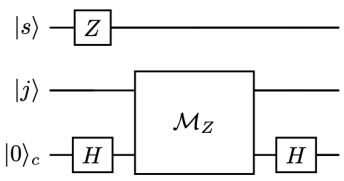


Figure 8: Block-encoding circuit for a type-C_n Cartan element.

Therefore,

$$(\langle 0|_c \otimes I_N) B_{H_C} (|0\rangle_c \otimes I_N) = \sigma_z \otimes \frac{\widetilde{Z}}{\alpha_Z} = \frac{\widetilde{H}_C}{\alpha_Z}.$$

Thus B_{H_C} is an $(\alpha_Z, 1, 0)$ block encoding of $\widetilde{H}_C$.

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Type D_n. Let s denote the internal qubit of each 2×2 block. Since

$$\widetilde{H}_D = H_D \oplus 0_{N-2n} = \widetilde{Z} \otimes J_0, \qquad J_0 = ZX,$$

define

$$B_{H_D} = B_{\widetilde{Z},(c,j)} \otimes (J_0)_s. \tag{107}$$

Thus, the circuit applies $B_{\widetilde{Z}}$ to (c, j) and applies X followed by Z to s, as shown in Figure 9.

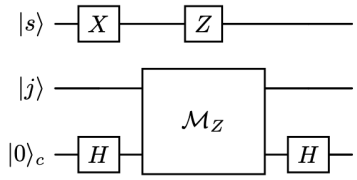


Figure 9: Block-encoding circuit for a type-D_n Cartan element.

Therefore,

$$\begin{aligned}(\langle 0|_c \otimes I_N) B_{H_D} (|0\rangle_c \otimes I_N) &= \frac{\widetilde{Z}}{\alpha_Z} \otimes J_0 \\ &= \frac{\widetilde{H}_D}{\alpha_Z}.\end{aligned}$$

Hence B_{H_D} is an exact $(\alpha_Z, 1, 0)$ block encoding of $\widetilde{H}_D$.

Corollary 4.1 (Circuit simplification for Cartan elements)**.** *At the level of the structured circuit blocks used throughout this paper, the Cartan construction reduces the number of parameter-dependent factors from $O(n^2)$ to $O(n)$ and eliminates the outer three-term, two-term, or eleven-term LCU assemblies used for generic elements of types A, B/D, and C, respectively.*

4.2 Block-Diagonal Subalgebras

4.2.1 Block Structure

Now we consider matrix Lie subalgebras whose elements consist of small diagonal blocks.

Let $N = mb$, and let $\mathfrak{g}_0$ be a matrix Lie algebra acting on $\mathbb{C}^b$. Define

$$\mathfrak{g} = \{A_1 \oplus \cdots \oplus A_m : A_j \in \mathfrak{g}_0\} \cong \mathfrak{g}_0^{\oplus m}. \tag{108}$$

Linear combinations preserve this block structure, and the Lie bracket is computed independently within each block:

$$\left[\bigoplus_{j=1}^m A_j, \bigoplus_{j=1}^m B_j\right] = \bigoplus_{j=1}^m [A_j, B_j]. \tag{109}$$

Thus $\mathfrak{g}$ is a Lie subalgebra.

For the unitary and orthogonal families, standard examples are

$$\mathfrak{su}(b)^{\oplus m} \subseteq \mathfrak{su}(N), \qquad \mathfrak{so}_b(\mathbb{C})^{\oplus m} \subseteq \mathfrak{so}_N(\mathbb{C}). \tag{110}$$

The symplectic family is treated similarly, with $b = 2r$. Writing $\Omega_r = \left(\begin{smallmatrix} 0 & I_r \\ -I_r & 0 \end{smallmatrix}\right)$, each local block satisfies $A_j^T \Omega_r + \Omega_r A_j = 0$. Consequently, their direct sum preserves $\Omega_r^{\oplus m}$ and realizes a subalgebra isomorphic to $\mathfrak{sp}_{2r}(\mathbb{C})^{\oplus m}$ inside $\mathfrak{sp}_{2rm}(\mathbb{C})$, expressed in a basis aligned with the symplectic blocks.

Under the identification $\mathbb{C}^N \cong \mathbb{C}^m \otimes \mathbb{C}^b$, an element of (108) takes the form

$$A = \sum_{j=1}^m |j\rangle\langle j| \otimes A_j. \tag{111}$$

Here the first factor labels the block, while the second contains its local coordinates.

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

4.2.2 Direct-Sum Block Encoding and Two-Level Factor Count

Let

$$A = \bigoplus_{j=1}^{m} A_j = \sum_{j=1}^{m} |j\rangle\langle j| \otimes A_j.$$

Suppose that U_j is an $(\alpha_j, a_j, 0)$ block encoding of A_j:

$$(\langle 0^{a_j}| \otimes I_b) U_j (|0^{a_j}\rangle \otimes I_b) = \frac{A_j}{\alpha_j}. \tag{112}$$

Each U_j is obtained from the family-specific construction developed earlier.

As a baseline construction, their ancillary spaces and normalization factors may be made common with an explicit scaling qubit. Set

$$a = \max_j a_j, \qquad \alpha \geq \max_j \alpha_j,$$

and pad each U_j to a ancillas. Introduce one additional qubit and define

$$\theta_j = \arccos\left(\frac{\alpha_j}{\alpha}\right), \qquad \widetilde{U}_j = R_y(2\theta_j) \otimes U_j.$$

Since $\langle 0|R_y(2\theta_j)|0\rangle = \alpha_j/\alpha$, the additional rotation rescales the encoded block from A_j/α_j to A_j/α. Then

$$(\langle 0^{a+1}| \otimes I_b) \widetilde{U}_j (|0^{a+1}\rangle \otimes I_b) = \frac{A_j}{\alpha}. \tag{113}$$

The global unitary is

$$U_A = \sum_{j=1}^{m} |j\rangle\langle j| \otimes \widetilde{U}_j. \tag{114}$$

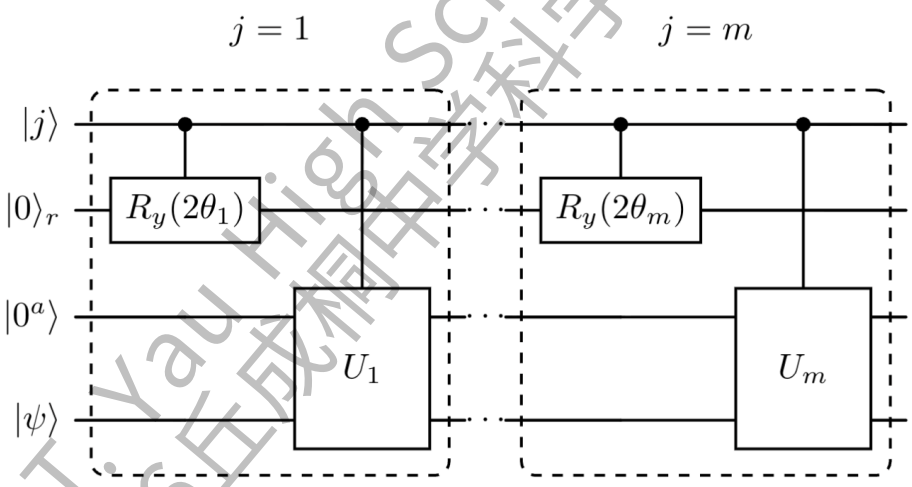


Figure 10: Baseline direct-sum block encoding with an explicit scaling qubit.

At register level, $|j\rangle$ denotes the block-label register, $|0\rangle_{\mathrm{r}}$ denotes the additional scaling qubit, $|0^a\rangle$ denotes the common padded ancillary register, and $|\psi\rangle \in \mathbb{C}^b$ is an arbitrary local system state. The branch labeled $j = k$ applies $R_y(2\theta_k) \otimes U_k = \widetilde{U}_k$. The control dots represent the projector $|k\rangle\langle k|$ on the complete block-label register.

Proposition 4.2 (Direct-sum block encoding)**.** *The unitary U_A defined above is an exact $(\alpha, a+1, 0)$-block encoding of A. The normalization factor may be chosen as $\alpha = \max_j \alpha_j$.*

Proof. Since the block-label subspaces are mutually orthogonal, $U_A^\dagger U_A = I$, and

$$(\langle 0^{a+1}| \otimes I_N) U_A (|0^{a+1}\rangle \otimes I_N) = \frac{1}{\alpha} \bigoplus_{j=1}^{m} A_j = \frac{A}{\alpha}. \tag{115}$$

The block label is a system register rather than an additional LCU selector; consequently, one may take $\alpha = \max_j \alpha_j$ instead of summing the normalizations over all blocks. □

2026 S.-T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Suppose that each standardized local encoding admits a two-level decomposition

$$\widetilde{U}_j = \prod_{\ell=1}^{L_j} F_{j,\ell}.$$

Lemma 4.3 (Block embedding of two-level unitaries)**.** *If F is a two-level unitary, then its embedding into the jth block,*

$$\mathcal{E}_j(F) = |j\rangle\langle j| \otimes F + \sum_{k\neq j} |k\rangle\langle k| \otimes I,$$

is also a two-level unitary.

Proof. If F mixes only two local basis states $|p\rangle$ and $|q\rangle$, then $\mathcal{E}_j(F)$ mixes only the two global states $|j,p\rangle$ and $|j,q\rangle$. Hence it is a two-level unitary. □

The global encoding can therefore be written as

$$U_A = \prod_{j=1}^{m} \prod_{\ell=1}^{L_j} \mathcal{E}_j(F_{j,\ell}).$$

Corollary 4.4 (Two-level factor count for block-diagonal matrices)**.** *If a complete local encoding of a $b \times b$ block requires at most $C(b)$ two-level factors, then*

$$L_{\text{total}} \leq \sum_{j=1}^{m} L_j \leq mC(b) = \frac{N}{b} C(b).$$

In particular, when b is fixed, the total factor count is $O(N)$.

The additional scaling qubit can be omitted when all local encodings already have the common normalization α. More generally, the diagonal and upper-triangular primitives above can be recompiled at larger normalizations without additional ancillas. Set $\alpha = \max_j \alpha_j$. For a local LCU encoding, let α_{jr} denote the normalization of its rth primitive and replace it by

$$\alpha'_{jr} = \frac{\alpha}{\alpha_j} \alpha_{jr}.$$

This common rescaling within each local encoding preserves the PREP probabilities and changes its total normalization from α_j to α. After padding the recompiled local unitaries U'_j to a ancillas, their direct sum satisfies

$$U'_A = \sum_{j=1}^{m} |j\rangle\langle j| \otimes U'_j, \qquad (\langle 0^a| \otimes I_N) U'_A (|0^a\rangle \otimes I_N) = \frac{A}{\alpha}.$$

Thus U'_A is an exact $(\alpha, a, 0)$ block encoding, saving one ancilla and eliminating the block-rescaling rotations. If all $\alpha_j = \alpha$, no recompilation is needed. All specialized resource counts reported below use U'_A; no separate numerical campaign is included for these specializations.

4.2.3 Repeated Blocks

A further simplification occurs when all local blocks are identical. In this case,

$$A = \underbrace{A_0 \oplus \cdots \oplus A_0}_{m \text{ copies}} = I_m \otimes A_0. \tag{116}$$

Corollary 4.5 (Repeated-block encoding)**.** *Let U_0 be an exact $(\alpha, a, 0)$-block encoding of A_0. Then the unitary*

$$U_A = I_m \otimes U_0 \tag{117}$$

is an exact $(\alpha, a, 0)$-block encoding of $A = I_m \otimes A_0$.

At register level, let $|j\rangle$ denote the block label and let $|\psi\rangle \in \mathbb{C}^b$ be an arbitrary local system state. Since every block uses the same encoding U_0, the block-label register is left unchanged, as shown in Figure 11. Projecting the block-encoding ancillas yields

$$(\langle 0^a| \otimes I_N) U_A (|0^a\rangle \otimes I_N) = I_m \otimes \frac{A_0}{\alpha} = \frac{A}{\alpha}.$$

2026 S. T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

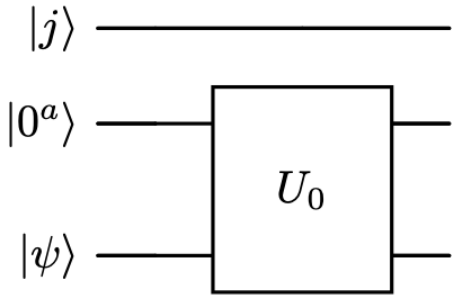


Figure 11: The repeated-block encoding.

Consequently, the same local circuit U_0 is reused for every block label, and no block-controlled selection is required.

More generally, partially repeated blocks may be written as

$$A = \bigoplus_{\lambda=1}^{s} \left(I_{m_\lambda} \otimes A_\lambda\right),$$

so that only the distinct local blocks A_λ need to be compiled separately.

5 Complexity of the Lie-Algebra-Aware Construction

Throughout this section, n denotes the Lie rank, d the original defining-representation dimension, and

$$q = \lceil \log_2 d \rceil, \quad N = 2^q \tag{118}$$

is the padded quantum-system dimension. The family dimensions are

$$\begin{array}{c|cccc} \text{family} & A_{n-1} & B_n & C_n & D_n \\ \hline d & n & 2n+1 & 2n & 2n \end{array}. \tag{119}$$

For type C_n, we additionally define

$$m_C = \frac{N}{2}, \tag{120}$$

whereas its original local block dimension is $n = d/2$. Every method uses the same embedding into dimension N. The analytical formulas below retain both d and N.

5.1 Resource conventions

Table 1 defines resources for one block-encoding call. Input-state preparation, measurement, and compilation of a nontrivial basis transformation are excluded. For an arbitrary input basis, the additional cost $2C(\widetilde{W})$ must be included.

The reported quantity G counts two-level factors after all operations are expanded on the full joint workspace. This convention applies uniformly to the primitive, SELECT, PREPARE, and UNPREPARE circuits.

In particular, let $\mathcal{W}$ denote the primitive workspace consisting of one block-encoding ancilla and the N-dimensional padded system, so that $\dim(\mathcal{W}) = 2N$. A selector-register rotation lifted to the complete circuit acts as

$$R_{ab} \otimes I_{\mathcal{W}} = \prod_{\mu=0}^{2N-1} G_{(a,\mu),(b,\mu)}(R_{ab}). \tag{121}$$

It therefore contributes $2N$ joint-space two-level factors. This is the same expansion rule used for register-local operations inside the diagonal and type-C primitives, and removes any dependence of the count on whether an operation is assigned to PREPARE, SELECT, or a primitive module.

For a selector-conditioned phase $\Phi_a(\phi) = I + (e^{i\phi} - 1)|a\rangle\langle a|$, its lift over the same workspace has the declared factorization

$$\Phi_a(\phi) \otimes I_{\mathcal{W}} = \prod_{j=0}^{N-1} G_{(a,2j),(a,2j+1)}\left(e^{i\phi} I_2\right). \tag{122}$$

2026 S. T. Yau High School Science Award
仅用于2026丘成桐中学科学奖论文公示

Table 1: Resource conventions used in the results.

Quantity	Meaning	Permitted comparison
G	Strict joint-space two-level count: primitive and SELECT schedules use their declared two-level expansions, and every outer PREPARE or UNPREPARE selector rotation is expanded over the complete $2N$-dimensional primitive workspace. Selector-conditioned phases are also expanded and included.	Compared across the Lie-aware, direct-isometry–Givens, and Halmos templates under this convention.
D	Safe serial depth of the same strict joint-space schedule. All two-level factors are serialized, so $D = G$ numerically for the declared templates. However, G counts factors, whereas D measures circuit depth.	Compared across the same templates under this convention.
n_{anc}	Encoding and LCU-selector ancillas.	Excludes system and structural registers.
α_{BE}	Complete block-encoding normalization.	Used in R_α, $\alpha_{\mathrm{BE}}G$, and $\alpha_{\mathrm{BE}}D$.

Thus each nontrivial selector phase contributes N joint-space two-level factors. These phase factors are included in G.

The quantity D is only a reproducible conservative depth for the compared templates, not a claim of minimum parallel depth or native hardware depth.

We compare all methods at the logical two-level-gate level because native-gate costs depend on hardware connectivity, available ancillas, synthesis accuracy, and compiler choices. Using the same logical counting rule makes the comparison reproducible. Under the same compilation method, a smaller G, and hence a smaller serial depth D, gives a lower conservative bound on native resources, although hardware-specific optimization may change the exact cost.

The primitive counts follow directly from the circuits constructed above. Consider first a generic, unpruned triangular primitive of original dimension r embedded into padded dimension s. By the elimination order in Eq. (49), column $j < r$ requires $j+1$ Givens rotations. Each of the remaining $s-r$ padded columns has a single unit entry in the Cholesky completion and therefore requires one additional completion rotation.

For a diagonal primitive of padded dimension s, Eq. (31) contains one address-controlled factor for each of the s system addresses. Moreover, each of the two outer layers $H_c \otimes I_s$ consists of s mutually disjoint two-level Hadamard factors under the convention of Section 5.1. In the purely imaginary construction, the explicit $Z_c \otimes I_s$ in Eq. (26) is absorbed into the s central address factors by replacing their local blocks with $\mathrm{diag}(e^{i\theta_j}, -e^{-i\theta_j})$. Hence it introduces no additional two-level factor beyond those s central factors, and

$$\begin{aligned} G_{\triangle}(r,s) &= \sum_{j=0}^{r-1}(j+1) + (s-r) = \frac{r(r+1)}{2} + (s-r), \\ G_{\mathrm{diag}}(s) &= s + s + s = 3s. \end{aligned} \tag{123}$$

In particular, $G_{\triangle}(s,s) = s(s+1)/2$. These are the generic unpruned template counts; structurally vanishing factors may be omitted in individual instances.

For a K-term LCU, coefficient-state synthesis uses

$$n_{\mathrm{PREPARE}} = K-1, \qquad n_{\mathrm{UNPREPARE}} = K-1 \tag{124}$$

selector-register coefficient rotations. After lifting each rotation over the $2N$-dimensional primitive workspace, the contributions to G are

$$G_{\mathrm{PREPARE}} = 2N(K-1), \qquad G_{\mathrm{UNPREPARE}} = 2N(K-1). \tag{125}$$

The family-specific values of K, the PREPARE and UNPREPARE rotation counts, and their contributions to G are summarized in Table 2.

5.2 Type A_{n-1}

For the three-term decomposition $D + T - T^\dagger$, the diagonal and triangular SELECT schedules contribute $3N + 2G_{\triangle}(d,N)$ two-level factors. Each of PREPARE and UNPREPARE contains $K-1=2$ selector rotations, and every such rotation expands into $2N$ joint-space two-level factors. The selector-conditioned minus sign

2026 S. T. Yau High School Science Award 仅用于2026丘成桐中学生数学奖论文公示

Table 2: Selector-register operations and their strict joint-space contributions in the generic, unpruned family decompositions.

Family	K	n_{PREPARE}	$n_{\mathrm{UNPREPARE}}$	$G_{\mathrm{PREPARE}} + G_{\mathrm{UNPREPARE}}$	SELECT phase factors
A	3	2	2	$8N$	N
B	2	1	1	$4N$	N
C	11	10	10	$40N$	N
D	2	1	1	$4N$	N

contributes a further N phase factors. Therefore,

$$\begin{aligned} G_A(d,N) &= 3N + 2G_{\triangle}(d,N) + 2\big[2N(3-1)\big] + N \\ &= d^2 - d + 14N, \\ D_A(d,N) &= G_A(d,N) = d^2 - d + 14N. \end{aligned} \tag{126}$$

If the diagonal branch vanishes, the active branch count is reduced to $K = 2$. In that case,

$$\begin{aligned} G_A^{(\mathrm{diag}=0)}(d,N) &= 2G_{\triangle}(d,N) + 2\big[2N(2-1)\big] + N = d^2 - d + 7N, \\ D_A^{(\mathrm{diag}=0)}(d,N) &= G_A^{(\mathrm{diag}=0)}(d,N) = d^2 - d + 7N. \end{aligned} \tag{127}$$

The normalization is

$$\Lambda_A = \alpha_D + 2\alpha_T, \qquad \alpha_D = \max_j |D_{jj}|, \qquad \alpha_T > \|T\|_2 . \tag{128}$$

The strict inequality makes $I - T^\dagger T/\alpha_T^2$ positive definite, so the ordinary Cholesky primitive is well defined.

5.3 Types B_n and D_n

For either orthogonal family, the original representation dimension is $d = 2n$ or $2n+1$, and the circuit acts on the padded dimension N. The two-term decomposition has $K = 2$. Its two triangular SELECT branches contribute $2G_{\triangle}(d,N)$ factors, while PREPARE and UNPREPARE together contribute $4N$, and the selector-conditioned minus sign contributes N phase factors. Hence,

$$\begin{aligned} G_O(d,N) &= 2G_{\triangle}(d,N) + 2\big[2N(2-1)\big] + N \\ &= d^2 - d + 7N, \\ D_O(d,N) &= G_O(d,N) = d^2 - d + 7N. \end{aligned} \tag{129}$$

Both families have

$$\Lambda_O = 2\alpha_T, \qquad \alpha_T > \|T\|_2 . \tag{130}$$

5.4 Type C_n

For C_n, the original representation dimension is $d = 2n$. After padding to dimension N, the local component matrices act on

$$m_C = \frac{N}{2}, \tag{131}$$

while their original dimension is $n = d/2$. The eleven-term implementation contains eight upper-triangular primitives, three diagonal primitives, nine nonidentity Pauli tensor factors, twenty selector-register coefficient rotations, and one selector phase.

After strict joint-space expansion, the eight triangular primitive schedules contribute $16G_{\triangle}(n, m_C)$. The three diagonal primitive schedules contribute $9N$, and the nine nonidentity Pauli tensor factors contribute a further $9N$. PREPARE and UNPREPARE together contribute $40N$, and the selector-conditioned minus sign contributes N phase factors. Therefore,

$$\begin{aligned} G_C(d,N) &= 16G_{\triangle}(n, m_C) + 18N + 40N + N \\ &= 2d^2 - 4d + 67N, \\ D_C(d,N) &= G_C(d,N) = 2d^2 - 4d + 67N. \end{aligned} \tag{132}$$

2026 S.-T. Yau High School Science Award 优胜奖 2026丘成桐中学科学奖论文公示

Using triangular parts T_K, T_S, T_P, T_Q and diagonal parts D_S, D_P, D_Q, the normalization is

$$\begin{aligned}\Lambda_C = \alpha_{T_K} + \alpha_{T_S} + \alpha_{T_P} + \alpha_{T_Q} \\ + \frac{1}{2}\left(\alpha_{D_S} + \alpha_{D_P} + \alpha_{D_Q}\right).\end{aligned} \tag{133}$$

Here each triangular normalization satisfies $\alpha_{T_Y} > \|T_Y\|_2$ for $Y \in \{K, S, P, Q\}$.

5.5 Overall scaling and native-gate upper bound

For generic dense members of all four families,

$$G_{\text{ours}} = \Theta(N^2), \qquad D_{\text{ours}} = \Theta(N^2). \tag{134}$$

Canonical Cartan inputs and fixed-width block-diagonal inputs are exceptions: their specialized constructions have $O(N)$ logical cost. We will introduce this below.

A Gray-path decomposition of one two-level unitary on $O(\log N)$ address qubits uses at most $O(\log^2 N)$ one- and two-qubit gates when sufficient work qubits are available [9]. Hence,

$$G_{\text{ours,native}} = O(N^2 \log^2 N), \qquad D_{\text{ours,native,seq}} = O(N^2 \log^2 N). \tag{135}$$

These formulas give conservative upper bounds, not exact hardware gate counts or depths. The logical block encoding constructed here is exact: the calculation of G and D does not discard nonzero entries, use low-rank approximations, or omit nonzero Lie components. When the circuit is compiled into a finite native gate set, each two-level unitary can be approximated to a chosen accuracy. If an approximate block encoding is allowed from the beginning, the cost may be further reduced by truncating small terms, using approximate factorizations, and distributing the error budget across the circuit. We leave these extensions for future work.

6 Competing Methods and Normalization-Weighted Cost

A generic dense unitary in $U(s)$ can be reduced to a diagonal unitary by the standard QR/Givens elimination into two-level unitaries [9]. Eliminating the entries below the diagonal requires

$$\sum_{j=1}^{s-1}(s-j) = \frac{s(s-1)}{2} \tag{136}$$

two-level Givens factors in the generic unpruned case. The remaining diagonal unitary contains $s-1$ independent relative phases after separating one overall phase. These relative phases can be implemented by $s-1$ two-level diagonal factors. We therefore define the generic dense-unitary template count

$$g_U(s) = \frac{s(s-1)}{2} + (s-1). \tag{137}$$

The separated overall phase fixes only the global representative of a standalone dense unitary and is not a separate physical factor in the comparator templates. If such a unitary is later used as a controlled branch, the induced relative phase must instead be included according to Eq. (122). Let

$$p = N - d \tag{138}$$

be the number of padded coordinates. Since $\widetilde{A} = A \oplus 0_p$, each padded zero coordinate forms an independent two-dimensional completion and contributes one two-level unitary. The following counts therefore exploit only the deterministic zero-padding structure, not any additional structure of A.

For Halmos dilation [8],

$$\begin{aligned}G_H(d,N) = D_H(d,N) &= g_U(2d) + (N-d) \\ &= 2d^2 + N - 1,\end{aligned} \qquad \alpha_H = \|A\|_2. \tag{139}$$

For direct isometry completion, Givens elimination of the physical $2d \times d$ isometry and the $N-d$ padded columns gives [9, 10]

$$\begin{aligned}G_{\text{iso}}(d,N) = D_{\text{iso}}(d,N) &= \sum_{j=1}^{d}(2d-j) + (N-d) \\ &= \frac{3d^2 - 3d + 2N}{2},\end{aligned} \qquad \alpha_{\text{iso}} = \|A\|_2. \tag{140}$$

2026 - Tsinghua University Science Award
仅用于2026年成都市学术学位公示

The two comparison methods are well-established block-encoding constructions that provide strong baselines. The stated counts are exact for the declared padding-aware templates, but are not claims of globally minimal circuits. Table 3 summarizes the closed-form expressions for G, D, and the block-encoding normalization for the proposed construction and the two comparison baselines.

Table 3: Generic explicit dense-matrix comparison.

Method	Cost per block-encoding call	Safe serial depth	Normalization
Type A_{n-1}, this work	$d^2 - d + 14N$ two-level units	$d^2 - d + 14N$	Λ_A
Type B_n, this work	$d^2 - d + 7N$ two-level units	$d^2 - d + 7N$	Λ_O
Type D_n, this work	$d^2 - d + 7N$ two-level units	$d^2 - d + 7N$	Λ_O
Type C_n, this work	$2d^2 - 4d + 67N$ two-level units	$2d^2 - 4d + 67N$	Λ_C
Direct isometry–Givens	$\frac{3d^2-3d+2N}{2}$ two-level units	$\frac{3d^2-3d+2N}{2}$	$\|A\|_2$
Direct Halmos	$2d^2 + N - 1$ two-level units	$2d^2 + N - 1$	$\|A\|_2$

For each method, we report

$$R_\alpha = \frac{\alpha_{\mathrm{BE}}}{\|A\|_2},$$

together with the single-call resources G, D and the normalization-weighted quantities $\alpha_{\mathrm{BE}}G, \alpha_{\mathrm{BE}}D$. These pairs characterize complementary costs. The unweighted quantities G and D are the direct compiler outputs for the declared joint-space two-level template of one block-encoding call. Reducing them lowers the size and safe serial depth of that template and reduces the number of logical synthesis tasks at this level. The weighted quantities additionally estimate the amplification of these costs in downstream algorithms whose query complexity depends on the block-encoding normalization.

The products $\alpha_{\mathrm{BE}}G$ and $\alpha_{\mathrm{BE}}D$ are informative query-weighted proxies, but they are not universal end-to-end objective functions. The actual cost also depends on the downstream algorithm, target accuracy, ancilla budget, depth constraints, and the available amplification procedure. Consequently, a method with the smallest weighted proxy is not automatically preferable in every implementation. We report all four quantities and interpret method selection as a resource tradeoff. In particular, reductions in G and D remain substantive compilation improvements even when another method has smaller $\alpha_{\mathrm{BE}}G$ or $\alpha_{\mathrm{BE}}D$.

7 Physical-Model Experiments

All models are generated directly in the canonical basis of their defining representations, so $W = I$. This choice isolates the block-encoding resources from the separate, input-dependent cost of compiling a basis transformation. For a nontrivial input basis, the construction remains valid by conjugation with $\widetilde{W}$, with an additional cost of $2C(\widetilde{W})$; this cost depends on the particular basis transformation and is therefore excluded from the present comparison. Every method receives the same canonical-basis input matrix.

The combined campaign contains 3380 valid matrices from R1–R4.The complete parameter grids, random generators, normalization policy, and validation statistics are given in Appendix A. All uncertainties are sample standard deviations. Direct isometry–Givens and Halmos resources are obtained from the preceding padding-aware expressions. All reported methods are compared through G, D, αG, and αD under the strict joint-space two-level convention of Section 5.1. These quantities are exact for the declared joint-space decompositions of the compared templates, but are not claims of globally minimal decompositions of the resulting unitaries.

Across the complete R1–R3 grids, the Lie-aware construction has both lower G and lower D than both dense baselines for 1060 of 1480 matrices (71.6%). The corresponding counts are 660/1000 in R1, 350/420 in R2, and 50/60 in R3. The reduction holds for every tested sample at $d \geq 32$ in R1, at $d \geq 16$ in R2, and at $d \geq 15$ in R3. Thus, the exceptions are confined to the smallest finite-dimensional regimes and do not alter the large-dimension trend predicted by the closed-form formulas.

Within the passive R4 subset, both G and D are lower than both baselines for all 270 samples at $d = 16, 32, 64$. Both normalization-weighted quantities are also lower than both baselines for all 180 passive samples at $d = 32, 64$.

7.1 R1: Three-Dimensional Random-Position Rydberg Networks

Frozen Rydberg gases provide an experimentally established platform for three-dimensional random hopping: atomic positions are disordered, while resonant dipole–dipole exchange transfers a single excitation through a long-range, anisotropic network [12]. Restricting the dynamics to the single-excitation sector produces a dense

2026 Siyuan High School Science Award 仅用于2026年度科学奖公示

$d \times d$ hopping matrix. A minimum separation models the preparation or blockade constraint and prevents unphysically large pair couplings.

Positions are sampled uniformly in a cubic or 2:1:1 box with $\|\mathbf{r}_i - \mathbf{r}_j\|_2 \geq r_{\min}$. The Hamiltonian and encoded generator are

$$H_{ij} = \frac{C_3\big(1 - 3\cos^2\theta_{ij}\big)}{\|\mathbf{r}_i - \mathbf{r}_j\|_2^3} \quad (i \neq j), \qquad H_{ii} = \epsilon_i, \qquad X = -i\left(H - \frac{\operatorname{tr} H}{d} I_d\right). \tag{141}$$

Thus $X^\dagger = -X$ and $\operatorname{tr} X = 0$, so $X \in \mathfrak{su}(d)$, the compact real form of type A_{d-1}. We use $C_3\varrho = 1$, $\epsilon_i = W\xi_i$ with $\xi_i \sim \mathcal{N}(0,1)$, and scan $d \in \{8, 16, 32, 64, 128\}$, $W/J \in \{0, 1, 4\}$, and $r_{\min} \in \{0.25, 0.35, 0.45\}$.

Table 4: R1 comparison at $N = 128$ for $(W/J, r_{\min}, \text{geometry}) = (1, 0.35, \text{cube})$ over 30 seeds. All rows use the strict joint-space two-level convention.

Method	R_α	G	D	αG	αD
Lie decomposition + primitive + LCU	1.977 ± 0.105	18048	18048	$1.114 \times 10^6 \pm 2.324 \times 10^5$	$1.114 \times 10^6 \pm 2.324 \times 10^5$
Direct isometry–Givens	1.000	24512	24512	$7.648 \times 10^5 \pm 1.530 \times 10^5$	$7.648 \times 10^5 \pm 1.530 \times 10^5$
Dense Halmos	1.000	32895	32895	$1.026 \times 10^6 \pm 2.054 \times 10^5$	$1.026 \times 10^6 \pm 2.054 \times 10^5$

The R1 portion of the experimental campaign is summarized in Table 4. The Lie-aware construction gives the smallest raw gate and depth counts for this experimentally motivated dense type-A model. Its larger normalization removes that advantage in αG and αD, for which direct isometry–Givens is preferable.

7.2 R2: Dense Quadratic Majorana/SYK$_2$ Networks

A general quadratic Hamiltonian of Majorana modes is specified completely by a real skew-symmetric coupling matrix. Such Hamiltonians describe free fermions, Bogoliubov–de Gennes systems, and networks of coupled Majorana modes. The all-to-all Gaussian ensemble is the quadratic SYK$_2$ model, a standard disordered-fermion model rather than an ad hoc random-matrix test [13]. It is useful here because it is dense and has no generic sparsity, low-rank, or translational structure that would favor an unrelated special-purpose encoding.

For $d = 2n$ Majorana operators satisfying $\{\gamma_\mu, \gamma_\nu\} = 2\delta_{\mu\nu}$, consider

$$\widehat{H}_{\mathrm{M}} = \frac{i}{4} \sum_{\mu,\nu=1}^{d} J_{\mu\nu}\gamma_\mu\gamma_\nu, \qquad J_{\mu\nu} \sim \mathcal{N}\left(0, \frac{J_0^2}{d}\right) \quad (\mu < \nu), \qquad J = -J^T. \tag{142}$$

The encoded object is the mode-space generator $X = J$, not the exponentially large Fock-space Hamiltonian. Since J is real and skew-symmetric, $X \in \mathfrak{so}(2n)$, the type-D_n family.

Table 5: R2 comparison for dense SYK$_2$ at $N = 256$ over 30 seeds. All rows use the strict joint-space two-level convention.

Method	R_α	G	D	αG	αD	Ancillas
Lie decomposition + primitive + LCU	1.665 ± 0.017	67072	67072	$2.179 \times 10^5 \pm 2.677 \times 10^3$	$2.179 \times 10^5 \pm 2.677 \times 10^3$	2
Direct isometry–Givens	1.000	98176	98176	$1.916 \times 10^5 \pm 2.179 \times 10^3$	$1.916 \times 10^5 \pm 2.179 \times 10^3$	1
Dense Halmos	1.000	131327	131327	$2.563 \times 10^5 \pm 2.914 \times 10^3$	$2.563 \times 10^5 \pm 2.914 \times 10^3$	1

The R2 portion of the experimental campaign is summarized in Table 5. For a dense, unstructured type-D ensemble, the Lie-aware route substantially reduces the raw logical circuit relative to both dense baselines. Its normalization makes direct isometry–Givens preferable in the weighted metrics, while the proposed construction remains better than Halmos.

7.3 R3: Odd-Dimensional Majorana Networks

Majorana networks may contain an effectively unpaired mode, for example in a defect, boundary, or subsystem description. The corresponding quadratic mode-space generator is an odd-dimensional real skew-symmetric matrix. Such a matrix necessarily has a zero eigenvalue, so this model probes a physically meaningful Majorana zero mode as well as the compiler's odd-dimensional branch. The odd-dimensional matrix should be interpreted as an effective mode-space sector; the computational padding coordinate is not the physical zero mode.

We use the same Gaussian couplings as Eq. (142), but with

$$d = 2n + 1 \in \{7, 15, 31, 63, 127, 255\}, \qquad J = -J^T, \qquad N = d + 1. \tag{143}$$

2026 S-T· You High School Science Award
仅用于2026年度科学奖论文公开

Table 6: R3 comparison at physical dimension $d = 255$, padded to $N = 256$, over 10 seeds. All rows use the strict joint-space two-level convention.

Method	R_α	G	D	αG	αD
Lie decomposition + primitive + LCU	1.663 ± 0.027	66562	66562	$2.156 \times 10^5 \pm 2.930 \times 10^3$	$2.156 \times 10^5 \pm 2.930 \times 10^3$
Direct isometry–Givens	1.000	97411	97411	$1.898 \times 10^5 \pm 1.707 \times 10^3$	$1.898 \times 10^5 \pm 1.707 \times 10^3$
Dense Halmos	1.000	130305	130305	$2.539 \times 10^5 \pm 2.284 \times 10^3$	$2.539 \times 10^5 \pm 2.284 \times 10^3$

Thus $J \in \mathfrak{so}(2n+1)$, the type-$B_n$ family, and $\operatorname{rank} J \leq d-1$. For the sampled dense matrices the maximal rank $d-1$ is attained, leaving one physical zero mode before padding.

The R3 portion of the experimental campaign is summarized in Table 6. The type-B route retains the raw-count advantage after odd-dimensional padding, confirming that the gain is not restricted to even-dimensional orthogonal generators. Direct isometry–Givens again wins the weighted metrics because of its optimal normalization.

7.4 R4: Multimode Bosonic Quadratic Systems

Quadratic bosonic Hamiltonians describe coupled harmonic modes, linear-optical mode conversion, squeezing, and Gaussian quantum dynamics [14, 15]. In canonical quadratures, the Heisenberg equations are generated by a real symplectic matrix. This makes multimode bosonic dynamics a direct physical realization of the type-C family rather than a generic non-Hermitian test matrix.

For n modes, let $R = (q_1, \ldots, q_n, p_1, \ldots, p_n)^T$ and

$$\widehat{H}_{\mathrm{B}} = \frac{1}{2} R^T K R, \qquad K = K^T > 0, \qquad X = J_{2n} K, \qquad J_{2n} = \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix}. \tag{144}$$

Because

$$X^T J_{2n} + J_{2n} X = 0, \tag{145}$$

$X \in \mathfrak{sp}(2n, \mathbb{R})$, the real type-$C_n$ algebra. The existing campaign uses $n \in \{2, 4, 8, 16, 32\}$, condition numbers $\kappa(K) \in \{2, 10, 50\}$, and combinations of dense conversion, squeezing, and rank-one common-bus coupling. Stability is enforced by $K > 0$ and the normalization $\|K\|_2 = \|X\|_2 = 1$.

Table 7: R4 comparison at $N = 64$ for weak squeezing, $\kappa(K) = 10$, and zero bus coupling over 30 seeds. All rows use the strict joint-space two-level convention.

Method	R_α	G	D	αG	αD
Lie decomposition + primitive + LCU	1.799 ± 0.056	12224	12224	$2.199 \times 10^4 \pm 6.822 \times 10^2$	$2.199 \times 10^4 \pm 6.822 \times 10^2$
Direct isometry–Givens	1.000	6112	6112	6112	6112
Dense Halmos	1.000	8255	8255	8255	8255

The R4 portion of the experimental campaign is summarized in Table 7. The conclusion changes in the passive, bus-dominated regime. In the campaign, "passive" denotes $(g_c, g_s) = (0.25, 0)$, so that the squeezing-related blocks vanish, while $\beta = 0.9$ makes the rank-one common-bus contribution dominant. Table 8 reports the existing $N = 64$ samples in this regime.

Table 8: R4 comparison at $N = 64$, $\kappa(K) = 2$, $(g_c, g_s) = (0.25, 0)$, and $\beta_{\mathrm{bus}} = 0.9$ over 10 seeds. Counts follow the strict joint-space two-level convention of Section 5.1.

Method	R_α	G	D	αG	αD
Lie decomposition + primitive + LCU	1.146 ± 0.009	3008	3008	$3.448 \times 10^3 \pm 2.717 \times 10^1$	$3.448 \times 10^3 \pm 2.717 \times 10^1$
Direct isometry–Givens	1.000	6112	6112	6112	6112
Dense Halmos	1.000	8255	8255	8255	8255

The two R4 tables show that the type-C comparison is parameter dependent. For the weak-squeezing central point, direct isometry–Givens has the smallest joint-space count and normalization-weighted cost. In the passive, bus-dominated regime, the Lie-aware construction eliminates the squeezing-related primitives and reduces both single-call resources G, D and both normalization-weighted quantities $\alpha G, \alpha D$ below both dense baselines. This identifies a physically motivated favorable regime, but does not imply superiority for generic type-C matrices.

2026 丘成桐中学数学论文奖 Yau High School Science Award

8 Scope of the Structured Specializations

The Cartan and block-diagonal constructions are retained as exact specializations of the unified compiler, rather than as claims of universal superiority over dedicated methods.

For Cartan inputs, the generic Lie–LCU assembly is bypassed, with

$$\alpha_{\mathrm{Car}} = \|X\|_2, \qquad G_{\mathrm{Car}} = D_{\mathrm{Car}} = \begin{cases} 3N, & A, \\ 4N, & B, C, D. \end{cases} \tag{146}$$

Hence the dispatcher reduces the generic quadratic count to $O(N)$ and eliminates the outer PREPARE and UNPREPARE circuits. This implementation is not optimal for diagonal type-A inputs. For $|a| \le 1$, the entrywise dilation

$$J(a) = \begin{pmatrix} a & \sqrt{1-|a|^2} \\ \sqrt{1-|a|^2} & -\overline{a} \end{pmatrix}$$

is unitary and has upper-left entry a. Applying this construction to the normalized diagonal entries requires only N two-level factors [8].

For

$$X = \bigoplus_{j=1}^{m} X_j, \tag{147}$$

the block label is a system register, so the local encodings can be combined with

$$\alpha_{\oplus} = \max_j \alpha_j \tag{148}$$

without an outer LCU. Recompiling the local primitives to this common normalization also removes the block-rescaling ancilla. The resource count is additive over the blocks,

$$G_{\oplus} = D_{\oplus} = \Theta\left(\sum_{j=1}^{m} b_j^2\right), \tag{149}$$

and becomes $O(N)$ for fixed block size.

For equal type-A blocks of size b, the present construction and a blockwise isometry–Givens encoding respectively require

$$G_{\oplus,A} = D_{\oplus,A} = N(b+13), \qquad G_{\text{block-iso}} = D_{\text{block-iso}} = \frac{N(3b-1)}{2}\,[10]. \tag{150}$$

Thus blockwise isometry is preferable for sufficiently small blocks, whereas the proposed route has a smaller raw count when

$$b > 27. \tag{151}$$

This work explores two specializations of the Lie-algebra-aware block-encoding framework: Cartan and block-diagonal inputs. The closed-form analysis shows that exploiting these structures can reduce the resource cost of the generic Lie–LCU construction. Although dedicated methods remain more efficient in some regimes or under certain resource measures, these results motivate further investigation of other special structures for which the present framework may offer useful advantages. Since the comparisons follow directly from the closed-form resource expressions, we do not include separate numerical experiments for these two specializations.

9 Conclusion and Discussion

We developed an explicit block-encoding framework for the defining representations of the four classical Lie families. It decomposes matrices into diagonal and strictly upper-triangular primitives, completes triangular contractions by a constructive Cholesky–Givens procedure, and combines them through family-specific LCU identities. In exact arithmetic, this gives exact block encodings with explicit normalizations and resource counts. It also exploits Cartan, block-diagonal, and direct-sum structure, using the maximum local normalization for direct sums.

The implementation validates the full construction, from decomposition and padding to LCU assembly and unitarity, with residuals at floating-point roundoff. The 3380 tests cover Rydberg, even- and odd-dimensional Majorana, and multimode bosonic models. Overall, the Lie-aware construction achieves lower logical gate counts and serial depths for the higher-dimensional type-A, B, and D instances, while its type-C advantage appears in specific structured regimes.

2026 Shing-Tung Yau High School Science Award
获得2026丘成桐中学科学奖论文公示

Across the dense type-A, B, and D models, the Lie-aware method shows a clear high-dimensional advantage: it has lower single-call counts than both dense baselines for all tests at $d \geq 32$ in R1, $d \geq 16$ in R2, and $d \geq 15$ in R3, and for 1060 of 1480 matrices overall. At the largest selected Majorana points (R2: $\rho = 0$, $d = 256$; R3: $d = 255$), it also retains lower normalization-weighted costs than dense Halmos. The type-C results identify a complementary favorable regime: all passive, bus-dominated tests at $d = 16, 32, 64$ have lower G, D than both dense baselines, and those at $d = 32, 64$ also have lower weighted costs. Together, these results establish concrete high-dimensional and structure-dependent regimes in which Lie-aware compilation provides a resource advantage, while normalization remains relevant when selecting among methods.

The method is therefore a structure-preserving fallback, not a replacement for specialized sparse, Toeplitz, circulant, low-rank, repeated-value, Cartan, or block-diagonal encodings. The counts apply only to the declared templates and are not global minima. The canonical-basis experiments exclude discovery, preprocessing, loading, basis changes, native-gate and hardware effects, fault tolerance, and hardware execution; the implementation is a research prototype. Higher-dimensional and exceptional representations, implicit oracles, approximate low-rank encodings, and complete downstream algorithms remain outside scope. Future work should combine Lie and additional structure, compile fixed-gate implementations with explicit error and hardware costs, and jointly optimize approximate encodings and downstream algorithms.

A Reproducibility of the Numerical Results

The accompanying code is a research prototype intended to reproduce the numerical validation and resource comparisons reported in this paper. It implements the canonical-basis decomposition, primitive and LCU validation, and resource accounting used in the R1–R4 experiments, but is not a production-ready implementation of the complete theoretical framework. Arbitrary basis-change compilation, native-gate lowering, data-loading circuits, and hardware-specific optimization are outside its present scope.

A.1 Numerical conventions

All methods use the same double-precision input matrices. Experiments were run with Python 3.11.7 and NumPy 1.26.4 using PCG64 seeds $s = 0, \ldots, S-1$, and spectral norms were computed by double-precision SVD. Nonzero triangular use

$$\alpha_T = (1 + 10^{-10})\|T\|_2 \tag{152}$$

while both baselines use $\alpha = \|X\|_2$. The factor $1 + 10^{-10}$ enforces the strict inequality $\alpha_T > \|T\|_2$, which ensures that $I - T^\dagger T/\alpha_T^2$ is positive definite for the Cholesky construction. This margin changes the triangular normalization only by a relative 10^{-10}, leaves G and D unchanged, and does not affect the reported conclusions. Lie-membership and circuit-validation tolerances are 10^{-10} and 10^{-8}, respectively. No nonzero matrix entry is truncated; 10^{-15} is used only to identify algebraically zero structural terms.

A.2 Campaign specification

Table 9 gives the complete R1–R4 campaign. Every noncentral grid point uses 10 seeds. The central points using 30 seeds are R1 $(W/J, r_{\min}, \text{geometry}) = (1, 0.35, \text{cube})$, R2 $\rho = 0$, and R4 $(g_c, g_s, \kappa, \beta) = (0.25, 0.10, 10, 0)$.

Table 9: Complete physical-model campaign. The number of parameter points is stated per dimension.

Model	Dimensions d	Parameter grid	Seeds per point	Matrices
R1	$8, 16, 32, 64, 128$	3 disorder values $\times$3 exclusions $\times$2 geometries	10; central 30	1000
R2	$8, 16, 32, 64, 128, 256$	$\rho \in \{0, 0.25, 0.5, 0.75, 1\}$	10; $\rho = 0$: 30	420
R3	$7, 15, 31, 63, 127, 255$	$\rho = 0$	10	60
R4	$4, 8, 16, 32, 64$	4 strength pairs $\times$3 condition numbers $\times$3 bus fractions	10; central 30	1900
			Total	3380

For R1, the number density is $\varrho = 1$, the mean-spacing and coupling units are

$$a = \varrho^{-1/3} = 1, \qquad J = \frac{C_3}{a^3} = C_3 \varrho = 1, \tag{153}$$

and $C_3 = 1$. The box volume is $V = d/\varrho = d$. For aspect vector $\mathbf{s} = (1, 1, 1)$ or $(2, 1, 1)$, its side lengths are $\mathbf{L} = (V/\prod_k s_k)^{1/3}\mathbf{s}$. Positions are sampled uniformly with rejection until $\|\mathbf{r}_i - \mathbf{r}_j\|_2 \geq r_{\min}$, where $r_{\min} \in \{0.25, 0.35, 0.45\}$. With quantization axis $\widehat{\mathbf{z}} = (0, 0, 1)$,

$$H_{ij} = \frac{1 - 3\cos^2\theta_{ij}}{\|\mathbf{r}_i - \mathbf{r}_j\|_2^3}, \quad \cos\theta_{ij} = \frac{(\mathbf{r}_j - \mathbf{r}_i)\cdot\widehat{\mathbf{z}}}{\|\mathbf{r}_j - \mathbf{r}_i\|_2}, \quad H_{ii} \sim \mathcal{N}(0, W^2), \tag{154}$$

2026 Yau High School Science Award
仅用于2026丘成桐科学奖论文公示

with $W/J \in \{0, 1, 4\}$ and $X = -i[H - \mathrm{tr}(H)I_d/d]$. No coupling cutoff is applied.

For R2, $J^{(r)}_{ij} \sim \mathcal{N}(0, 1/d)$ independently for $i < j$ and $J^{(r)} = -J^{(r)T}$. Define $\ell = \max(1, d/4)$ and $J^{(s)}_{i,i+k} = e^{-k/\ell} = -J^{(s)}_{i+k,i}$, then rescale $J^{(s)}$ so that $\|J^{(s)}\|_F = \|J^{(r)}\|_F$. The encoded matrix is

$$X = \sqrt{1-\rho^2}\, J^{(r)} + \rho J^{(s)}, \qquad \rho \in \{0, 0.25, 0.5, 0.75, 1\}. \tag{155}$$

R3 uses the same generator with odd d and $\rho = 0$.

For R4, put $m = d/2$, draw $\omega_i \sim U[1, 2]$, independent standard-normal matrices Z_K, Z_S, Z_C, and a normalized standard-normal vector u. With

$$M = \frac{Z_K + Z_K^T}{\sqrt{2m}}, \quad S = \frac{g_s(Z_S + Z_S^T)}{\sqrt{2m}}, \quad C = \frac{g_s Z_C}{\sqrt{m}}, \quad B = g_c\left(\sqrt{1-\beta^2}M + \beta u u^T\right), \tag{156}$$

form

$$K_0 = \begin{pmatrix} \mathrm{diag}(\omega) + B + S & C \\ C^T & \mathrm{diag}(\omega) + B - S \end{pmatrix}. \tag{157}$$

Define

$$\delta = \max\{0,\, 0.25 - \lambda_{\min}(K_0)\}, \qquad \overline{K}_0 = K_0 + \delta I_d, \tag{158}$$

and let $\lambda_- = \lambda_{\min}(\overline{K}_0)$ and $\lambda_+ = \lambda_{\max}(\overline{K}_0)$. The rescaled matrix used in the experiment is

$$K = \frac{1}{\kappa} I_d + \left(1 - \frac{1}{\kappa}\right) \frac{\overline{K}_0 - \lambda_- I_d}{\lambda_+ - \lambda_-}. \tag{159}$$

If $\lambda_+ = \lambda_-$, we instead set $K = I_d$. Thus $\lambda_{\min}(K) = 1/\kappa$, $\lambda_{\max}(K) = 1$, and $\|K\|_2 = 1$. Finally, $X = J_d K$. The grid is

$$(g_c, g_s) \in \{(0.25, 0), (0.25, 0.10), (0.25, 0.25), (0.10, 0.25)\}, \quad \kappa \in \{2, 10, 50\}, \quad \beta \in \{0, 0.5, 0.9\}. \tag{160}$$

A.3 Correctness evidence

We validate the complete LCU construction for all 3380 R1–R4 matrices. Table 10 reports mean and maximum residuals. Here $\widetilde{X}$ denotes the padded input. All cases passed the 10^{-10} membership/decomposition and 10^{-8} circuit-validation thresholds.

Table 10: Mean and maximum residuals over the 3380 R1–R4 matrices.

Diagnostic	Mean	Maximum
Lie-membership residual	5.19×10^{-16}	2.84×10^{-14}
Structural-decomposition residual	4.32×10^{-17}	1.28×10^{-16}
Primitive projected-block residual	7.49×10^{-16}	6.03×10^{-15}
PREPARE first-column residual	3.70×10^{-17}	3.33×10^{-16}
Projected block residual $\|\,\mathrm{block}(U) - \widetilde{X}/\Lambda\|_2$	3.36×10^{-16}	3.07×10^{-15}
Global unitarity residual $\|U^\dagger U - I\|_2$	2.78×10^{-15}	2.13×10^{-14}

2026 S. T. Yau High School Science Award
仅用于2026丘成桐中学科学奖公示

References

[1] R. Wiersema, E. Kökcü, A. F. Kemper, and B. N. Bakalov, "Classification of dynamical Lie algebras of 2-local spin systems on linear, circular and fully connected topologies," *npj Quantum Information*, vol. 10, Art. 110, 2024. doi: 10.1038/s41534-024-00900-2.

[2] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, "Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pp. 193–204, 2019. doi: 10.1145/3313276.3316366.

[3] G. H. Low and I. L. Chuang, "Hamiltonian simulation by qubitization," *Quantum*, vol. 3, Art. 163, 2019. doi: 10.22331/q-2019-07-12-163.

[4] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, "Simulating Hamiltonian dynamics with a truncated Taylor series," *Physical Review Letters*, vol. 114, Art. 090502, 2015. doi: 10.1103/PhysRevLett.114.090502.

[5] D. Camps, L. Lin, R. Van Beeumen, and C. Yang, "Explicit quantum circuits for block encodings of certain sparse matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 45, no. 1, pp. 801–827, 2024. doi: 10.1137/22M1484298.

[6] C. Sünderhauf, E. Campbell, and J. Camps, "Block-encoding structured matrices for data input in quantum computing," *Quantum*, vol. 8, Art. 1226, 2024. doi: 10.22331/q-2024-01-11-1226.

[7] C. Yang, Z. Li, H. Yao, Z. Fan, G. Zhang, and J. Liu, "Dictionary-based block encoding of sparse matrices with low subnormalization and circuit depth," *Quantum*, vol. 9, Art. 1805, 2025. doi: 10.22331/q-2025-07-22-1805.

[8] P. R. Halmos, "Normal dilations and extensions of operators," in *Selecta*. New York, NY, USA: Springer, pp. 113–122, 1983. Available: Semantic Scholar.

[9] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *Physical Review A*, vol. 52, no. 5, pp. 3457–3467, 1995. doi:10.1103/PhysRevA.52.3457.

[10] R. Iten, R. Colbeck, I. Kukuljan, J. Home, and M. Christandl, "Quantum circuits for isometries," *Physical Review A*, vol. 93, Art. 032318, 2016. doi:10.1103/PhysRevA.93.032318.

[11] J. E. Humphreys, *Introduction to Lie Algebras and Representation Theory*, Graduate Texts in Mathematics, vol. 9. New York: Springer, 1972. doi:10.1007/978-1-4612-6398-2.

[12] C. Lippe, T. Klas, J. Bender, P. Mischke, T. Niederprum, and H. Ott, "Experimental realization of a 3D random hopping model," *Nature Communications*, vol. 12, 6976, 2021. doi:10.1038/s41467-021-27243-2.

[13] P. Łydżba, M. Rigol, and L. Vidmar, "Entanglement in many-body eigenstates of quantum-chaotic quadratic Hamiltonians," *Physical Review B*, vol. 103, 104206, 2021. doi:10.1103/PhysRevB.103.104206.

[14] Arvind, B. Dutta, N. Mukunda, and R. Simon, "The real symplectic groups in quantum mechanics and optics," *Pramana*, vol. 45, pp. 471–497, 1995. arXiv:quant-ph/9509002.

[15] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, "Gaussian quantum information," *Reviews of Modern Physics*, vol. 84, pp. 621–669, 2012. doi:10.1103/RevModPhys.84.621.

2026 Tianyu High School Science Award 仅用于2026年度高中科学奖论文公示