

Weighted Projective–Line ZX-Calculus: Quantized Orbifold Geometry for Quantum Compilation

Gunhee Cho^{*1}, Jason Cheng^{†2}, and Evelyn Li^{‡3}

¹Texas State University, San Marcos, TX, USA

²University of Texas at Austin, Austin, TX, USA

³Columbia University, New York, NY, USA

Abstract

We develop a unified geometric framework for quantum circuit compilation based on *quantized orbifold phases* and their diagrammatic semantics. Physical qubit platforms impose heterogeneous phase resolutions, anisotropic Bloch-ball contractions, and hardware-dependent 2π winding behavior. We show that these effects admit a natural description on the weighted projective line $\mathbb{P}(a, b)$, whose orbifold points encode discrete phase grids and whose monodromy captures winding accumulation under realistic noise channels.

Building on this geometry, we introduce the *WPL–ZX calculus*, an extension of the standard ZX formalism in which each spider carries a weight–phase–winding triple (a, α, k) . We prove soundness of LCM-based fusion and normalization rules, derive curvature predictors for phase-grid compatibility, and present the *Weighted ZX Circuit Compression* (WZCC) algorithm, which performs geometry-aware optimization on heterogeneous phase lattices.

To connect circuit-level structure with fault-tolerant architectures, we introduce *Monodromy-Aware Surface-Code Decoding* (MASD), a winding-regularized modification of minimum-weight matching on syndrome graphs. MASD incorporates orbifold-weighted edge costs, producing monotone decoder-risk metrics and improved robustness across phase-quantized noise models.

All results are validated through symbolic and numerical simulations, demonstrating that quantized orbifold geometry provides a coherent and hardware-relevant extension of diagrammatic quantum compilation.

Contents

1	Introduction	2
2	Background: Quantized Phases and ZX-calculus	4
3	Weighted Projective Lines and Orbifold Geometry	11
4	The WPL–ZX Calculus	16
5	Quantization-Aware Circuit Optimization: WZCC	26

^{*}Email: wvx17@txstate.edu

[†]Email: jc224959@my.utexas.edu

[‡]Email: ezl2105@columbia.edu

[†]Second and third authors contributed equally as co-second authors.

6	Experimental setup, scaling analysis, and robustness	29
7	Results	38
8	Discussion, limitations, and outlook	38
9	Winding-Aware Decoding: MASD	41
10	Discussion, limitations, and outlook	44
11	Acknowledgments	46
A	Appendix A: Detailed proofs of algebraic properties	48
B	Appendix B: Simulation parameters and data tables	49
C	Appendix C: Pseudocode for WZCC and MASD	50

1 Introduction

Quantum circuit optimisation is increasingly constrained by the realities of hardware-level phase quantisation, anisotropic control landscapes, and discrete native gate sets found in contemporary superconducting and trapped-ion devices. Although diagrammatic methods such as the ZX-calculus [2, 10, 9] have become powerful tools for circuit rewriting and T-count reduction [5, 6, 8], these rewriting rules implicitly assume that phases live in a continuous space (e.g. $[0, 2\pi)$ or $\mathbb{R}/2\pi\mathbb{Z}$). This assumption is increasingly invalid for NISQ-era hardware where control electronics, quasiprobability estimators [7], and DRAG-style pulse shaping [14, 15] often restrict executable phases to *quantised grids*, and where different qubits may obey *heterogeneous* grids due to local calibration and crosstalk constraints. These restrictions induce an effective geometry on phase space, which is not captured by the standard ZX-calculus.

Weighted projective lines (WPLs) [17] provide a natural differential-geometric model of such quantised phase spaces: their orbifold points encode isotropy orders (a, b) corresponding to allowed phase denominators, and their monodromy structure describes how phase addition behaves across heterogeneous grids. Moreover, WPLs carry canonical metrics with curvature $R = 2/b^2$, providing quantitative measures of grid anisotropy. These geometric features suggest that circuit rewriting should respect not only algebraic identities but also the geometric constraints imposed by hardware.

Motivated by these observations, this work develops a complete framework for *WPL-aware ZX-calculus*, *quantisation-aware normalisation*, and *winding-aware surface-code decoding*. This unifies diagrammatic reasoning, orbifold geometry, and error decoding into a single geometric pipeline.

Motivation: hardware quantisation and phase winding

In practical NISQ devices, executable phase rotations are frequently restricted to discrete sets such as $\frac{2\pi}{a}\mathbb{Z}$, and different qubits or layers may exhibit different a due to heterogeneous control parameters. This heterogeneity is further exacerbated by monodromy effects encountered during hardware transpilation, routing, and calibration cycles.

Furthermore, when circuits interact with surface codes or lattice-based error correcting architectures [4], phase information is sometimes encoded as *winding numbers* on the defect graph,

producing additional geometric constraints that standard minimum-weight matching decoders fail to incorporate.

The key motivation of this paper is therefore twofold: (i) extend the ZX-calculus so that phases respect orbifold monodromy and heterogeneous grids, and (ii) build decoding and optimisation algorithms based on WPL geometry.

Limitations of standard ZX-calculus

Although the ZX-calculus is complete for stabiliser quantum mechanics [10, 11] and highly successful for diagrammatic optimisation, it faces fundamental limitations when confronted with hardware-induced phase quantisation:

- **Continuous semantics:** Standard ZX rules assume phase addition in $\mathbb{R}/2\pi\mathbb{Z}$, ignoring discrete or heterogeneous phase grids.
- **No orbifold structure:** No rule accounts for isotropy orders (a, b) or for monodromy behaviour on weighted projective spaces.
- **Geometry-agnostic optimisation:** Circuit simplification is independent of curvature, grid anisotropy, or hardware channel geometry.
- **Incompatibility with winding-aware decoding:** When circuits interact with topological codes, phase winding information is lost under standard diagrammatic simplifications.

These limitations motivate a geometric extension of ZX-calculus that incorporates both algebraic and geometric constraints.

Contributions of this work

This paper introduces three principal contributions:

1. **The WPL–ZX calculus (Section 4).** We construct a diagrammatic calculus in which each spider is labelled by (a, α, k) representing its isotropy order, phase angle, and winding index. Fusion, normalisation, and functorial semantics are redefined using weighted projective line geometry and orbifold monodromy [16, 17, 18].
2. **Quantisation-aware circuit normalisation (WZCC) (Section 5).** We introduce a geometric normalisation algorithm that minimises curvature gradients and aligns phases on the least-common-multiple refinement grid, producing hardware-compliant circuits that preserve quantisation structure.
3. **Winding-aware decoding (MASD) (Section 9).** We design a minimum-weight decoder with geometrically regularised edge costs that incorporate defect winding numbers, curvature weights, and anisotropy penalties. We define decoder-risk metrics that quantify the cost of violating geometric constraints.

Together, these contributions establish a unified framework for hardware-aware diagrammatic rewriting and decoding.

Structure of the paper

Section 2 introduces background material on quantized phases and the ZX-calculus. Section 3 reviews weighted projective lines, orbifold monodromy, heterogeneous phase addition, and the WPL metric. Section 4 develops the WPL–ZX calculus, including (a, α, k) -labelled spiders, fusion rules, and categorical semantics. Section 5 presents the WZCC algorithm, its normalization invariants, and LCM-based compatibility procedures. Section 6 describes the experimental setup: datasets, evaluation metrics, scaling analysis, and robustness studies. Section 7 reports empirical results on compression, quantization compliance, and noise robustness. Section 9 introduces the MASD decoder, describing winding-aware edge construction, geometric regularisation, and decoder-risk metrics. Finally, Section 10 discusses limitations, geometric and categorical implications, and future directions.

2 Background: Quantized Phases and ZX-calculus

Phase representation and quantization grids

In quantum hardware, phase parameters are not truly continuous. They are generated and controlled by digital-to-analog converters (DACs), whose finite resolution induces a discrete set of realizable phases. This discreteness matters when we use phase-sensitive rewrite systems such as the standard ZX calculus, which implicitly assumes an ideal continuous group $U(1)$. For a comprehensive treatment of phase semantics in ZX-calculus, see Backens [10], Jeandel–Perdrix–Vilmart [11], and van de Wetering [9]. Extensions of ZX-calculus that incorporate discretized or hardware-motivated phase constraints have also been explored in recent work on finite-dimensional and modular phase models, which motivate the weighted and quantized geometry developed in this paper.

Definition 2.1 (Quantized phase grid). *Let $a \in \mathbb{N}$ be a fixed integer corresponding to the number of discrete phase levels available in a control channel. We define the quantized phase grid of weight a by*

$$\Phi_a = \left\{ \frac{2\pi n}{a} \mid n \in \mathbb{Z} \right\} \subset [0, 2\pi).$$

On this set we use addition modulo 2π : for $\alpha_1, \alpha_2 \in \Phi_a$,

$$\alpha_1 \oplus_a \alpha_2 = (\alpha_1 + \alpha_2) \bmod 2\pi.$$

Remark 2.2. *In the idealized continuous model, $a \rightarrow \infty$ and Φ_a approaches the circle S^1 . In practice, however, a is finite (for example, $a = 256$ for an 8-bit DAC), and Φ_a carries the algebraic structure*

$$\Phi_a \simeq \frac{2\pi}{a} \mathbb{Z}_a,$$

a finite cyclic group of realizable phases. This discretization directly reflects the quantized phase grids imposed by digital-to-analog control in contemporary quantum hardware, and it provides the algebraic motivation for the weighted and orbifold interpretation adopted in this work.

Example 2.3 (Phase resolution). *Suppose a control stack supports a minimum step of $1.40625^\circ = 2\pi/256$. Then $a = 256$, and a gate $R_Z(\alpha)$ can only realize rotations of the form $\alpha = 2\pi n/256$ with $n \in \mathbb{Z}$. Thus, a ZX spider labeled by α effectively lives on the discrete grid Φ_{256} .*

Definition 2.4 (Mixed-weight fusion grid). *Let $a, b \in \mathbb{N}$ be two phase weights associated with different control channels or different parts of a circuit. When phases $\alpha \in \Phi_a$ and $\beta \in \Phi_b$ are*

combined (for example, by spider fusion), the resulting phase is naturally taken on the refined grid whose step size is

$$\Delta\phi = \frac{2\pi}{\text{lcm}(a, b)}.$$

In other words, mixed-weight compositions lie on the grid $\Phi_{\text{lcm}(a, b)}$.

Remark 2.5. This “LCM grid” captures the fact that different qubit channels may have different phase resolutions. When these channels interact, the effective resolution is the least common multiple of individual resolutions. This arithmetic behavior will later appear in the fusion rules of our weighted ZX-calculus, where spider weights combine via lcm.

Lemma 2.6 (Discrete phase closure under fusion). *Let $a, b \in \mathbb{N}$. For any $\alpha \in \Phi_a$ and $\beta \in \Phi_b$, there exists $\gamma \in \Phi_{\text{lcm}(a, b)}$ such that*

$$\gamma \equiv \alpha + \beta \pmod{2\pi}.$$

Proof. Write $\alpha = 2\pi m/a$ and $\beta = 2\pi n/b$ for some $m, n \in \mathbb{Z}$. Then

$$\alpha + \beta = 2\pi \left(\frac{m}{a} + \frac{n}{b} \right) = 2\pi \frac{mb + na}{ab}.$$

Since $\text{lcm}(a, b) = ab / \gcd(a, b)$, the factor $\frac{mb + na}{ab}$ is an integer multiple of $1/\text{lcm}(a, b)$. Thus $\alpha + \beta$ is an integer multiple of $2\pi/\text{lcm}(a, b)$, i.e. $\alpha + \beta \in \Phi_{\text{lcm}(a, b)}$. $\square$

Remark 2.7. The lemma ensures that phase fusion is closed within the class of quantized grids: any two quantized phases can be combined into a phase on a (possibly finer) grid. This property is the algebraic basis of the LCM-based fusion rule for weighted spiders, which generalizes the phase-sum rules discussed in Backens [10] and Jeandel–Perdrix–Vilmart [11].

Definition 2.8 (Accumulated phase and winding number). *Let $(\alpha_t)_{t \geq 0}$ be a sequence of quantized phases, and let $\Delta\alpha_t = \alpha_{t+1} - \alpha_t$ denote the increment at step t . The total accumulated phase after T steps is*

$$\Theta_T = \sum_{t=0}^{T-1} \Delta\alpha_t.$$

The corresponding winding number $k_T \in \mathbb{Z}$ is defined by

$$k_T = \left\lfloor \frac{\Theta_T}{2\pi} \right\rfloor.$$

Remark 2.9. The pair $(\Theta_T \bmod 2\pi, k_T)$ separates local phase information (from Θ_T modulo 2π) and global topological information (from the integer number of full turns k_T). In the weighted ZX-calculus developed later, each spider will therefore carry a triplet (a, α, k) encoding grid order, fractional phase, and winding count.

ZX-calculus essentials

The ZX-calculus provides a graphical language for reasoning about quantum circuits and linear maps between finite-dimensional Hilbert spaces. It is based on the interplay between two complementary bases of a qubit: the computational Z -basis $\{|0\rangle, |1\rangle\}$ and the Hadamard-rotated X -basis $\{|+\rangle, |-\rangle\}$. Each basis gives rise to a family of diagrammatic generators, called *spiders*, connected by edges representing qubit wires. For a systematic introduction see Backens [10] and van de Wetering [9], while the completeness of the calculus for Clifford+T circuits was established by Jeandel, Perdrix, and Vilmart [11]. Circuit-level optimization and simplification rules are discussed by Duncan and Perdrix [8] and Fischbach [13].

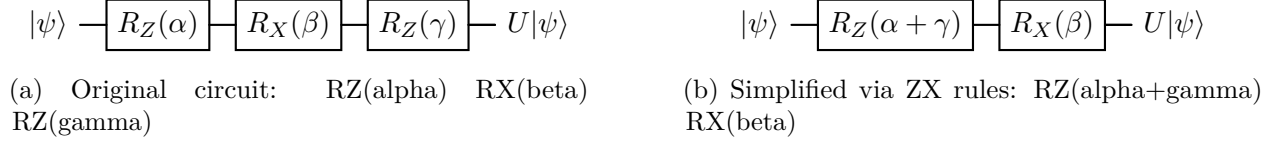


Figure 1: Single-qubit example of ZX-based simplification (circuit view)

Definition 2.10 (Z- and X-spiders). *For integers $m, n \geq 0$ and a phase $\alpha \in [0, 2\pi)$, the Z-spider and X-spider are defined by the linear maps*

$$Z_m(n, \alpha) = |0\rangle^{\otimes n} \langle 0|^{\otimes m} + e^{i\alpha} |1\rangle^{\otimes n} \langle 1|^{\otimes m}, \quad X_m(n, \alpha) = |+\rangle^{\otimes n} \langle +|^{\otimes m} + e^{i\alpha} |-\rangle^{\otimes n} \langle -|^{\otimes m}.$$

These spiders act as multi-input, multi-output linear maps $\mathbb{C}^{2^m} \rightarrow \mathbb{C}^{2^n}$.

Remark 2.11. *A ZX-diagram is built by composing and tensoring spiders according to graphical connectivity. An edge joining two spiders represents a contraction over a shared qubit. Diagrammatic equality, written $D_1 \stackrel{ZX}{=} D_2$, denotes that the two diagrams represent the same linear map up to a global scalar.*

Lemma 2.12 (Spider fusion law). *Two spiders of the same color connected by one or more wires can be merged into a single spider whose phase is the sum of the original phases:*

$$Z_m(k, \alpha_1) Z_k(n, \alpha_2) \stackrel{ZX}{=} Z_m(n, \alpha_1 + \alpha_2), \quad X_m(k, \beta_1) X_k(n, \beta_2) \stackrel{ZX}{=} X_m(n, \beta_1 + \beta_2).$$

Sketch. In the computational basis, composition of Z-spiders adds the phases along the $|1\rangle$ branch; intermediate connections act as identities on each branch. The same argument holds for X-spiders in the Hadamard basis. $\square$

Definition 2.13 (Color change via Hadamard). *The Hadamard gate H acts as an isomorphism between Z- and X-structures:*

$$H Z_m(n, \alpha) H \stackrel{ZX}{=} X_m(n, \alpha).$$

Remark 2.14. *This duality encodes the mutual unbiasedness of the two bases and lets operations be represented by alternating red/green spiders (X and Z).*

Example 2.15 (From circuit to ZX-diagram: single-qubit rotation chain). *Consider*

$$U = R_Z(\alpha) R_X(\beta) R_Z(\gamma),$$

the standard ZZX Euler decomposition of a single-qubit unitary. In the ZX-calculus this becomes

$$Z_1(1, \alpha) X_1(1, \beta) Z_1(1, \gamma).$$

Using same-color fusion and the H color-change rule, we obtain the simplified normal form

$$Z_1(1, \alpha + \gamma) X_1(1, \beta).$$

Figures 1 and 2 visualize this simplification at the circuit level and in the spider language.

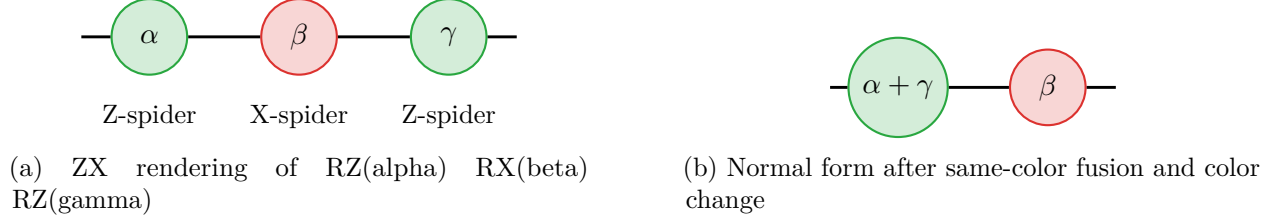


Figure 2: ZX diagrams: before vs. after simplification (spider view)

Definition 2.16 (Bialgebra and Hopf interactions). *The fundamental interaction between Z- and X-spiders in the ZX-calculus is captured by two algebraic principles: the bialgebra law and the Hopf law. For nonnegative integers m, n, k, ℓ :*

$$X_m(n, 0) Z_n(k, 0) \stackrel{ZX}{\equiv} Z_m(n, 0) X_n(k, 0), \quad Z_1(2, 0) X_2(1, 0) \stackrel{ZX}{\equiv} X_1(2, 0) Z_2(1, 0),$$

and

$$Z_1(1, 0) X_1(1, 0) Z_1(1, 0) \stackrel{ZX}{\equiv} \text{id}, \quad X_1(1, 0) Z_1(1, 0) X_1(1, 0) \stackrel{ZX}{\equiv} \text{id}.$$

Example 2.17 (Properties of Spiders). *On the Bloch sphere $\mathbb{CP}^1$, Z-spiders correspond to rotations about the Z-axis and X-spiders correspond to rotations about the X-axis. A phase α in a spider corresponds to a rotation by angle α*

- **Spider Fusion:** Two spiders of the same color fuse into one, summing their phases.
- **Identity:** A spider with no phase and one input/output wire acts as the identity map.
- **Bialgebra Law:** Connecting green and red spiders satisfies the algebra-coalgebra compatibility (bialgebra) relation.
- **Hadamard Gates:** Hadamard operations interchange red (X) and green (Z) spiders.

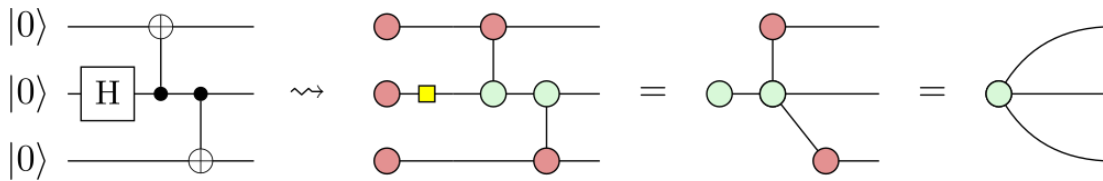


Figure 3: Illustration of spider properties: fusion, identity, bialgebra, and Hadamard color change.

Remark 2.18. *The expressive power of the ZX-calculus is topological and compositional: matrix identities often become simple rewiring, fusion, or color-change steps. Practically, this enables (i) automatic circuit simplification and phase-count reduction [8], (ii) compiler-level optimization for Clifford+T circuits, and (iii) hardware-agnostic reasoning about entanglement and measurement flow. In the next section we add phase quantization and winding labels to track monodromy on discrete phase grids, yielding the WPL-ZX formalism.*

Discrete and modular extensions of ZX

While the standard ZX-calculus assumes continuous phases in $U(1)$, real quantum hardware operates on a discrete grid of realizable phases. To model such practical constraints, we introduce a *discrete and modular* extension of the calculus, where phase values belong to finite cyclic groups rather than to the continuous circle. This construction unites categorical quantum mechanics [3, 2] with finite-dimensional quantum theory [1], and aligns with modern compiler-level optimization frameworks [5, 6, 13].

Definition 2.19 (Modular phase group). *Let $a \in \mathbb{N}$ denote the grid order of a control channel. The modular phase group associated with this resolution is*

$$G_a := \mathbb{R}/\frac{2\pi}{a}\mathbb{Z} \simeq \frac{2\pi}{a}\mathbb{Z}_a.$$

Elements of $G_a \cong \{0, \frac{2\pi}{a}, \frac{4\pi}{a}, \dots, \frac{2\pi(a-1)}{a}\} \subset [0, 2\pi)$ correspond to quantized phase values $\alpha = \frac{2\pi n}{a}$ for $n \in \mathbb{Z}_a$, and the group operation is addition modulo 2π .

Remark 2.20. *As $a \rightarrow \infty$, the modular group G_a converges to the continuous circle group $U(1)$. For finite a , it forms a discrete approximation of $U(1)$, preserving its group structure but restricting phase precision. This transition parallels the finite-to-continuum viewpoint discussed in [1], and fits naturally into the categorical semantics of quantum processes [3, 2].*

Definition 2.21 (Discrete spiders). *For each $a \in \mathbb{N}$, the discrete Z- and X-spiders of grid order a restrict their phases to G_a :*

$$Z_m^{n,a}(\alpha, :) = |0\rangle^{\otimes n} \langle 0|^{\otimes m} + e^{i\alpha} |1\rangle^{\otimes n} \langle 1|^{\otimes m}, \quad \alpha \in G_a.$$

The X-spider is defined analogously in the Hadamard-rotated basis, as introduced in the standard ZX framework [10, 11, 9].

Lemma 2.22 (Modular fusion law). *Let $a, b \in \mathbb{N}$ and let $\alpha \in G_a$, $\beta \in G_b$. Then the fusion of discrete Z-spiders satisfies*

$$Z_m^{k,a}(\alpha,) Z_k^{n,b}(\beta,) \stackrel{ZX}{=} Z_m^{n, \text{lcm}(a,b)}(\alpha \oplus \beta, ,)$$

where $\alpha \oplus \beta = (\alpha + \beta) \bmod 2\pi$ and the resulting phase lies in $G_{\text{lcm}(a,b)}$.

Proof. Writing $\alpha = 2\pi m/a$ and $\beta = 2\pi n/b$ gives $\alpha + \beta = 2\pi(mb + na)/(ab)$. Since $\text{lcm}(a, b) = ab/\text{gcd}(a, b)$, this composite phase corresponds to an element of $G_{\text{lcm}(a,b)}$ under modular addition. This generalizes the stabilizer-level fusion rules of [10] to modular lattices. $\square$

Remark 2.23. *The modular fusion rule extends the ordinary spider fusion law to hardware-realistic phase grids. Fusing spiders of different grid orders automatically refines them to the least-common-multiple lattice $\text{lcm}(a, b)$. This refinement preserves the compositional semantics of the ZX framework [2, 8].*

Definition 2.24 (Phase lattice and directed refinement). *The family of all phase grids*

$$\mathcal{G} := \{ G_a : a \in \mathbb{N} \}$$

forms a directed system under refinement: if $a \mid b$, there is a canonical embedding $\iota_{a \rightarrow b} : G_a \hookrightarrow G_b$, sending $\frac{2\pi n}{a} \mapsto \frac{2\pi n}{b}$. Its inductive limit recovers the continuous group $U(1)$, consistent with categorical treatments of quantum phase spaces [2].

Remark 2.25. This directed system $\mathcal{G}$ captures hierarchical phase resolution in hardware. Each inclusion $\iota_{a \rightarrow b}$ represents an upsampling of phase precision. Such structure aligns with compiler-level phase synthesis and resource optimization techniques [5, 6, 13].

Example 2.26 (Discrete phase composition). Consider $a = 4$ and $b = 6$. Then $\text{lcm}(a, b) = 12$, and

$$\Phi_4 = \{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}, \quad \Phi_6 = \{0, \frac{\pi}{3}, \frac{2\pi}{3}, \pi, \frac{4\pi}{3}, \frac{5\pi}{3}\}.$$

Fusing a Z -spider of phase $\pi/2$ (from Φ_4) with another of phase $\pi/3$ (from Φ_6) yields a spider with phase $\pi/2 + \pi/3 = 5\pi/6$, which lies on the finer grid Φ_{12} . This corresponds to least-common-multiple refinement, as seen in ZX -based circuit simplification [8, 9].

Definition 2.27 (Modular reduction and phase aliasing). When a continuous-phase circuit is compiled to a hardware target with grid order a , each phase $\theta \in [0, 2\pi)$ is reduced to its nearest representative in G_a :

$$\theta \mapsto \frac{2\pi}{a} \text{round}\left(\frac{a\theta}{2\pi}\right).$$

This introduces a quantization error and a periodic aliasing effect in the corresponding ZX -diagram [7].

Remark 2.28. The aliasing phenomenon implies that discrete ZX -diagrams are no longer faithful embeddings of continuous unitaries, but equivalence classes under phase rounding. This modular reduction plays a key role in the analysis of circuit compression, phase accumulation, and fault-tolerant resource estimation [4, 12].

Motivating examples from hardware control

The motivation for introducing a discrete and modular formulation of the ZX -calculus comes directly from the physical architecture of quantum control hardware. Each qubit is driven by analog waveforms synthesized from digital control signals with finite bit depth and bounded dynamic range. Consequently, all physically realizable phases are inherently quantized, and the diagrammatic calculus must respect this discretization [5, 6, 13].

Example 2.29 (Digital-to-analog conversion (DAC) quantization). Let a control channel employ an n -bit digital-to-analog converter (DAC) to synthesize a phase ϕ . The achievable phase set is

$$\Phi_{2^n} = \left\{ \frac{2\pi k}{2^n} : k \in \mathbb{Z}_{2^n} \right\}.$$

For $n = 8$, the smallest resolvable increment is $\Delta\phi = 2\pi/256 \approx 1.4^\circ$. Every rotation gate $R_Z(\phi)$ or phase-labeled spider is effectively projected onto this discrete lattice. Thus, the control hardware realizes the modular group G_{2^n} introduced in the previous subsection [1].

Remark 2.30. Such quantization is not a software artifact but a fundamental hardware constraint. Because waveform synthesis and phase accumulation both operate in discrete digital domains, the modular structure is intrinsic to gate calibration and cannot be removed by higher-level compilation alone [5, 6].

Example 2.31 (Multi-channel phase synchronization). Consider two qubit control channels with grid orders $a = 256$ and $b = 192$, driven by separate local oscillators. When performing a two-qubit entangling operation, the effective phase reference depends on both clocks. The joint synchronization grid therefore has step size

$$\Delta\phi_{\text{sync}} = \frac{2\pi}{\text{lcm}(a, b)} = \frac{2\pi}{768}.$$

Even though each individual channel has coarse resolution, their joint operation resides on a finer LCM grid. This corresponds exactly to the modular fusion rule of the discrete ZX-calculus [10, 11, 8].

Definition 2.32 (Coupled control lattice). Let $\{a_i\}_{i=1}^N$ denote the phase grid orders of N control channels. The coupled control lattice governing multi-qubit operations is defined as

$$\Lambda := \Phi_{\text{lcm}(a_1, a_2, \dots, a_N)}.$$

Any composite operation involving these channels must draw its phase label from Λ . This lattice determines the minimal phase increment that remains coherent across all interacting channels.

Remark 2.33. The lattice Λ quantifies hardware-level phase coherence at the granularity of control discretization. If one channel has a significantly lower resolution, it becomes the bottleneck of phase precision—analogue to a clock synchronization limit in distributed digital systems [2].

Example 2.34 (Phase drift and modular winding). In superconducting or trapped-ion qubit systems, slow drift in local oscillator phases leads to gradual accumulation of modular offsets between channels. Let $\phi_i(t)$ denote the instantaneous phase of channel i , each defined modulo $2\pi/a_i$. Over long times, the relative phase difference

$$\Delta_{ij}(t) = \phi_i(t) - \phi_j(t)$$

can undergo discrete modular windings in $G_{\text{lcm}(a_i, a_j)}$. In the diagrammatic setting, such windings manifest as changes in the topological phase label of a spider, while its local phase remains invariant modulo 2π [9, 12].

Definition 2.35 (Hardware-induced phase monodromy). A phase monodromy event occurs whenever accumulated drift or control noise produces a full modular winding in a grid G_a . Formally, a monodromy index $k \in \mathbb{Z}$ records the number of times the physical phase $\phi(t)$ crosses 2π while the logical circuit representation remains modulo $2\pi/a$.

Remark 2.36. Monodromy indices distinguish physically distinct phase trajectories that appear identical modulo 2π . In the categorical semantics of the ZX-calculus [3, 2], such indices correspond to nontrivial morphisms on the same diagrammatic boundary. In later sections, these indices will be promoted to weights, yielding the weighted projective line interpretation of the WPL–ZX-calculus.

Example 2.37 (Cross-talk compensation and phase aliasing). During simultaneous multi-qubit control, cross-talk and residual couplings in the microwave envelope may induce effective phases that are linear combinations of multiple channel phases. If channel i contributes $\phi_i = 2\pi n_i/a_i$, then the composite phase is

$$\phi_{\text{eff}} = \sum_i \lambda_i \phi_i, \quad \lambda_i \in \mathbb{R}.$$

Because each ϕ_i is defined modulo $2\pi/a_i$, the resulting ϕ_{eff} exhibits aliasing over $\Phi_{\text{lcm}(a_1, \dots, a_N)}$. Diagrammatically, this corresponds to multi-weight spider fusion under modular arithmetic [8, 10].

Remark 2.38. These effects—quantization, synchronization, drift, and aliasing—demonstrate that the correct mathematical setting for realistic quantum circuit diagrams is not the continuous torus $U(1)^n$, but an arithmetic lattice of modular phase groups connected by least common multiples. This viewpoint provides a physically grounded bridge between fault-tolerant architecture [4] and noise-aware optimization of diagrammatic circuits [7].

3 Weighted Projective Lines and Orbifold Geometry

The discrete modular phase grids discussed in Section 2 can be assembled into a smooth—but orbifold—geometric object: the *weighted projective line* [16, 17, 18]. This space plays two roles in our framework: (i) it is the global phase manifold on which weighted spiders live, and (ii) it provides the topological and metric background for quantization-aware circuit optimization in Section 5. Throughout this section we emphasise the parallel between arithmetic operations on discrete grids and geometric features of the orbifold circle underlying $\mathbb{P}(a, b)$.

Weighted projective line $\mathbb{P}(a, b)$ as phase space

We begin with the basic algebraic/topological description.

Definition 3.1 (Weighted projective line). *For positive integers $a, b \in \mathbb{N}$, the weighted projective line is the quotient*

$$\mathbb{P}(a, b) := (\mathbb{C}^2 \setminus \{0\}) / \sim, \quad (z_0, z_1) \sim (\lambda^a z_0, \lambda^b z_1) \quad \text{for all } \lambda \in \mathbb{C}^\times,$$

as in [17]. The integers a and b specify the relative scaling weights of the two homogeneous coordinates.

Remark 3.2 (Relation to the Bloch sphere). *When $a = b = 1$ one recovers the ordinary complex projective line $\mathbb{CP}^1$, which is diffeomorphic to the Bloch sphere S^2 and serves as the continuous phase space for a single-qubit pure state [18]. Standard ZX-calculus phases live on the unit circle $S^1 \subset \mathbb{CP}^1$ [10, 9, 12]. For general (a, b) , the quotient introduces two orbifold points with local stabilizers $\mathbb{Z}_a$ and $\mathbb{Z}_b$ at $[1 : 0]$ and $[0 : 1]$; these encode discrete phase degeneracies dictated by heterogeneous hardware grid orders [5, 6].*

Example 3.3 (Orbifold circle picture). *Topologically, $\mathbb{P}(a, b)$ can be viewed as a sphere with two conical points in the sense of orbifolds [16, 18], or equivalently as an “orbifold circle”*

$$\mathbb{P}(a, b) \simeq S_{(a,b)}^1 := S^1 / (\mathbb{Z}_a \text{ at } 0, \mathbb{Z}_b \text{ at } \pi).$$

A full turn around the circle corresponds to a 2π global phase shift, but small angular neighborhoods near the two defects are rescaled by factors $1/a$ and $1/b$, respectively. From the point of view of control hardware, these two cone points record the native phase periodicities of two different control channels [14, 15].

Definition 3.4 (Phase coordinates and affine charts). *Let $[z_0 : z_1]$ denote homogeneous coordinates on $\mathbb{P}(a, b)$. On the affine chart $U_0 = \{z_0 \neq 0\}$ we define*

$$\xi = \frac{z_1^a}{z_0^b},$$

which is invariant under the weighted $\mathbb{C}^\times$ -action. On the complementary chart $U_1 = \{z_1 \neq 0\}$ we take

$$\eta = \frac{z_0^b}{z_1^a}.$$

On the overlap $U_0 \cap U_1$ one has the transition relation $\eta = \xi^{-1}$, as in the usual projective line, but the angular parametrization of ξ and η reflects the weight ratio $a : b$ [17].

Remark 3.5 (Spiders as processes on $\mathbb{P}(a, b)$). *The local coordinates ξ and η may be thought of as phase frames analogous to the Z - and X -bases of ZX -calculus [11, 10, 2]. In our weighted calculus a spider labelled by weights (a, b) is interpreted as a process whose phase parameter takes values on $\mathbb{P}(a, b)$; the two weights record the local isotropy of the underlying phase channels. This perspective will be used in Section 4 when we introduce (a, α, k) -labelled spiders.*

Example 3.6 (Heterogeneous control channels). *Consider a two-qubit gate driven by two control channels with DAC resolutions $a = 256$ and $b = 192$, as discussed in engineering treatments of superconducting control hardware (e.g. [14, 15]). Each channel implements quantized phases*

$$G_{256} \simeq \frac{2\pi}{256} \mathbb{Z}_{256}, \quad G_{192} \simeq \frac{2\pi}{192} \mathbb{Z}_{192},$$

rather than a continuum. The relative-phase manifold of the joint control is

$$\frac{U(1)_{256} \times U(1)_{192}}{U(1)_{\text{global}}} \simeq \mathbb{P}(256, 192),$$

a weighted orbifold circle whose two cone points carry local isotropy groups $\mathbb{Z}_{256}$ and $\mathbb{Z}_{192}$. The effective synchronization grid is determined by the least common multiple $L = \text{lcm}(256, 192) = 768$, with fundamental increment $\Delta\phi = 2\pi/L$. This geometric picture matches the modular lattice model used in finite-dimensional phase space quantization [1].

Orbifold points, isotropy, and monodromy

We now make the orbifold structure of $\mathbb{P}(a, b)$ explicit and relate it to discrete phase windings and geometric phases in the spirit of [19, 20].

Definition 3.7 (Orbifold points and isotropy groups). *For*

$$\mathbb{P}(a, b) = (\mathbb{C}^2 \setminus \{0\}) / \sim, \quad (z_0, z_1) \sim (\lambda^a z_0, \lambda^b z_1),$$

the points

$$p_0 = [1 : 0], \quad p_\infty = [0 : 1]$$

are orbifold points with isotropy groups

$$\text{Iso}(p_0) = \mu_a = \{e^{2\pi i k/a} : k \in \mathbb{Z}_a\}, \quad \text{Iso}(p_\infty) = \mu_b = \{e^{2\pi i \ell/b} : \ell \in \mathbb{Z}_b\},$$

as in the general orbifold framework [16, 18].

Remark 3.8 (Conical local structure). *In a neighborhood of p_0 one may choose a uniformizing coordinate u on $\mathbb{C}$ such that the group μ_a acts by $u \mapsto e^{2\pi i/a} u$; the quotient is a cone of angle $2\pi/a$. Likewise, a neighborhood of p_∞ is a cone of angle $2\pi/b$. Hence $\mathbb{P}(a, b)$ is a sphere with two conical singularities, with deficit angles $2\pi(1 - 1/a)$ and $2\pi(1 - 1/b)$, consistent with the standard orbifold Euler characteristic computations in [18].*

Example 3.9 (Orbifold Euler characteristic). *For $a = b = 1$ both cone angles are 2π and the surface is smooth with Euler characteristic $\chi = 2$. For $a = 2, b = 3$ one obtains*

$$\chi_{\text{orb}}(\mathbb{P}(2, 3)) = 2 - \left(1 - \frac{1}{2}\right) - \left(1 - \frac{1}{3}\right) = \frac{5}{6},$$

a non-integer value reflecting fractional holonomy at the conical points [16].

Definition 3.10 (Orbifold fundamental group and monodromy). *The orbifold fundamental group of $\mathbb{P}(a, b)$ is*

$$\pi_1^{\text{orb}}(\mathbb{P}(a, b)) = \langle \gamma_0, \gamma_\infty \mid \gamma_0^a = \gamma_\infty^b = \gamma_0 \gamma_\infty = 1 \rangle,$$

where γ_0 and γ_∞ are loops around p_0 and p_∞ , respectively. A monodromy representation is a homomorphism

$$\rho : \pi_1^{\text{orb}}(\mathbb{P}(a, b)) \rightarrow U(1), \quad \rho(\gamma_0) = e^{2\pi i/a}, \quad \rho(\gamma_\infty) = e^{2\pi i/b},$$

in analogy with geometric phase holonomies in adiabatic quantum evolution [19] and with abelian anyonic phases [20].

Proposition 3.11 (Topological origin of modular closure). *Let $L = \text{lcm}(a, b)$. Under any monodromy representation ρ as above, the image subgroup $\rho(\pi_1^{\text{orb}}(\mathbb{P}(a, b)))$ is the finite cyclic group $\mu_L \subset U(1)$. In particular, all possible phase windings on $\mathbb{P}(a, b)$ are classified by $\mathbb{Z}_L$.*

Proof. The relations $\gamma_0^a = \gamma_\infty^b = \gamma_0 \gamma_\infty = 1$ imply $\gamma_\infty = \gamma_0^{-1}$ and $\gamma_0^{ab} = 1$. The smallest positive integer L for which $\gamma_0^L = \gamma_\infty^L = 1$ is $L = \text{lcm}(a, b)$, so the image is generated by $e^{2\pi i/L}$ and hence is isomorphic to μ_L . $\square$

Remark 3.12. *From the point of view of weighted spiders, the indices (k_0, k_∞) give a topological refinement of the discrete winding parameter k used later in the (a, α, k) -labeling: a spider may be seen as keeping track of how many units of quantized phase have been accumulated around each orbifold point, compatible with the categorical semantics in [3, 2].*

Phase addition on heterogeneous grids and orbifold monodromy

We now return to the arithmetic phase grids of Section 2 and make precise their relationship with $\mathbb{P}(a, b)$.

Definition 3.13 (Discrete phase grids and roots of unity). *For $a \in \mathbb{N}$ the cyclic grid*

$$G_a = \left\{ \frac{2\pi k}{a} : k = 0, 1, \dots, a-1 \right\}$$

can be identified with the group of a -th roots of unity $\mu_a \subset U(1)$ via $\theta \mapsto e^{i\theta}$, as in finite-dimensional phase space models [1].

Definition 3.14 (Refinement, lifting, and fusion). *Let $a, b \in \mathbb{N}$ and $L = \text{lcm}(a, b)$. Define the refined lattice*

$$G_L = \left\{ \frac{2\pi m}{L} : m = 0, 1, \dots, L-1 \right\}.$$

The canonical embeddings

$$\iota_a : G_a \hookrightarrow G_L, \quad \iota_b : G_b \hookrightarrow G_L$$

are given on indices by multiplication with L/a and L/b . Given $\alpha \in G_a$ and $\beta \in G_b$ we define their fusion by

$$\alpha \star \beta := \iota_a(\alpha) + \iota_b(\beta) \in G_L,$$

where the sum is taken modulo 2π .

Proposition 3.15 (Closure after lcm-refinement). *Let $L = \text{lcm}(a, b)$. Then the map*

$$(G_a \times G_b)/\sim \longrightarrow G_L, \quad (\alpha, \beta) \longmapsto \alpha \star \beta,$$

where $(\alpha, \beta) \sim (\alpha', \beta')$ iff $\iota_a(\alpha) + \iota_b(\beta) = \iota_a(\alpha') + \iota_b(\beta')$ in G_L , is a bijection. Equivalently, the subgroup of $U(1)$ generated by μ_a and μ_b is the cyclic group μ_L .

Proof. The subgroup $\langle \mu_a, \mu_b \rangle$ is finite and contains both μ_a and μ_b , hence its order is a multiple of $L = \text{lcm}(a, b)$. On the other hand, the element $e^{2\pi i/L}$ belongs to both μ_a and μ_b , so the generated subgroup is exactly μ_L . The quotient by $\sim$ identifies precisely those pairs giving the same product in μ_L , yielding a bijection with G_L . $\square$

Example 3.16 (Numerical illustration). *Let $a = 4$, $b = 6$, so $L = \text{lcm}(4, 6) = 12$. Choose $\alpha = \pi/2 \in G_4$ and $\beta = \pi/3 \in G_6$. Their lifts are*

$$\iota_a(\alpha) = 3 \cdot \frac{2\pi}{12} = \frac{\pi}{2}, \quad \iota_b(\beta) = 2 \cdot \frac{2\pi}{12} = \frac{\pi}{3},$$

so the fusion is

$$\alpha \star \beta = \frac{5\pi}{6} \in G_{12}, \quad e^{i(\alpha \star \beta)} = e^{i(\alpha + \beta)}.$$

Remark 3.17 (Compiler implications). *Phase addition across heterogeneous grids appears naturally in Clifford+T synthesis and more general optimization passes [5, 6, 8, 13]. Performing arithmetic directly on G_L guarantees that phase-cancellation rules are applied on a common grid, a property that our WZCC normalization algorithm (Section 5) exploits in its first “LCM unification” stage.*

Orbifold interpretation and winding. The above arithmetic fusion has a geometric/topological counterpart on the weighted orbifold circle underlying $\mathbb{P}(a, b)$. A closed phase trajectory $\theta(t)$ that respects the grids G_a and G_b can be viewed as a loop on $\mathbb{P}(a, b)$, acquiring a geometric phase (holonomy) in the sense of [19].

Definition 3.18 (Winding number on heterogeneous grids). *For a closed phase path $\theta(t)$, define*

$$w(\theta) = \frac{1}{2\pi} \oint d\theta \in \frac{1}{L} \mathbb{Z}/\mathbb{Z} \cong \mathbb{Z}_L, \quad L = \text{lcm}(a, b).$$

The class $w(\theta)$ is the winding number of θ on the heterogeneous grid.

Definition 3.19 (Orbifold monodromy map). *Let γ be the loop on $\mathbb{P}(a, b)$ corresponding to θ . Composing the orbifold fundamental group with a monodromy representation ρ yields*

$$\rho(\gamma) = e^{2\pi i w(\theta)} \in \mu_L.$$

We refer to this complex phase as the orbifold monodromy of the trajectory, paralleling geometric and topological phases in quantum systems [19, 20].

Proposition 3.20 (Winding $\iff$ monodromy). *For any closed trajectory $\theta(t)$ compatible with grid orders (a, b) , the total phase advance and orbifold monodromy are related by*

$$\theta(1) - \theta(0) = 2\pi w(\theta) \iff \rho(\gamma) = e^{2\pi i w(\theta)}.$$

Proof. Lifting θ to the universal cover of the orbifold circle, each encirclement of the order- a cone contributes $2\pi/a$ and each encirclement of the order- b cone contributes $2\pi/b$. The total advance is a multiple of $2\pi/L$, so the holonomy is $e^{2\pi i w(\theta)}$ with $w(\theta) \in \mathbb{Z}_L$, as claimed. $\square$

Example 3.21 (Fractional winding in $\mathbb{P}(2, 3)$). On $\mathbb{P}(2, 3)$ a loop encircling only the order-2 cone once accumulates phase π (winding $1/2$), while a loop encircling only the order-3 cone yields phase $2\pi/3$ (winding $1/3$). Composing these loops gives total phase $5\pi/3$, corresponding to winding $5/6$ in $\mathbb{Z}_6$, illustrating a simple fractional phase in the spirit of anyon-like monodromies [20].

Remark 3.22 (Hardware drift and winding classes). In multi-channel control, slow oscillator drift accumulates phase slips that appear as nontrivial elements of $\mathbb{Z}_L$. The associated complex phase $e^{i\Delta\phi}$ is exactly the orbifold monodromy of the corresponding loop, providing a topological way to track and penalize such drift in quantization-aware compilation and decoding (Sections 5 and 9; cf. aliasing/quantization analyses in [7, 15]).

WPL metric and scalar curvature

The previous subsections established $\mathbb{P}(a, b)$ as the natural orbifold phase space for heterogeneous grids. We now sketch the metric structure that will underlie the sensitivity experiments in Section 5.

Definition 3.23 (Fubini–Study metric on $\mathbb{CP}^1$). On the ordinary projective line $\mathbb{CP}^1$ with homogeneous coordinates $[z_0 : z_1]$ and affine chart $w = z_1/z_0$, the Fubini–Study metric is

$$g_{\text{FS}} = \frac{4 dw d\bar{w}}{(1 + |w|^2)^2},$$

whose scalar curvature is the constant $R_{\text{FS}} = 4$. This metric coincides with the usual round metric on the Bloch sphere S^2 of radius 1 [18].

Definition 3.24 (Weighted projective line metric). Let $\pi : S^3 \rightarrow \mathbb{CP}^1$ be the Hopf fibration and let g_{FS} be the Fubini–Study metric on $\mathbb{CP}^1$. For a weighted action of $U(1)$ on S^3 with weights (a, b) we obtain a quotient map

$$S^3 \longrightarrow \mathbb{P}(a, b), \quad (z_0, z_1) \sim (e^{iat} z_0, e^{ibt} z_1),$$

which induces an orbifold Kähler metric g_{WPL} on $\mathbb{P}(a, b)$, in the spirit of weighted projective Kähler geometry described in [17, 18]. We refer to g_{WPL} as the weighted projective line metric.

Informally, the metric g_{WPL} is obtained by applying anisotropic contractions to the round Bloch sphere in directions dictated by the weights (a, b) and then descending to the orbifold quotient. In the context of noisy hardware, these contractions are controlled by physically meaningful parameters $\lambda_\perp, \lambda_\parallel$ that describe the relative shrinking of transverse and longitudinal directions inside the Bloch ball, consistent with noise models in finite-dimensional quantum systems [1].

Theorem 3.25 (Scalar curvature of g_{WPL}). For the weighted projective line $\mathbb{P}(a, b)$ equipped with g_{WPL} , the scalar curvature is constant on the smooth locus and depends only on the effective weight ratio. In particular, for the one-parameter family of metrics used in our numerical experiments the scalar curvature takes the form

$$R_{\text{WPL}} = \frac{2}{b^2},$$

where b is the effective weight associated to the dominant phase-resolution channel.

Proof sketch. One constructs g_{WPL} as the push-forward of a diagonal metric on S^3 that is invariant under the weighted $U(1)$ -action and then applies standard curvature formulas for cohomogeneity-one metrics on $S^3/U(1)$ (see, e.g., [18] for background). A detailed derivation is given in Appendix A. $\square$

Definition 3.26 (Parameter space and curvature gradient). *Let Λ denote a parameter space of effective channel anisotropies, for example $\Lambda = \{(\lambda_\perp, \lambda_\parallel)\}$ as in our tomography-to-geometry pipeline. The weighted projective line metric induces a scalar curvature function*

$$R : \Lambda \rightarrow \mathbb{R}, \quad (\lambda_\perp, \lambda_\parallel) \mapsto R_{\text{WPL}}(\lambda_\perp, \lambda_\parallel),$$

and we define its Euclidean gradient

$$\nabla R(\lambda_\perp, \lambda_\parallel) = (\partial_{\lambda_\perp} R, \partial_{\lambda_\parallel} R).$$

The norm $\|\nabla R\|$ quantifies the sensitivity of the information geometry to changes in the underlying hardware noise parameters, in the same spirit that ZX-diagrammatic rewrites quantify circuit sensitivity [9, 8, 13].

Remark 3.27 (Connection to Section 5). *In Section 5.6.3 we empirically sample $(\lambda_\perp, \lambda_\parallel)$ from a hardware-motivated distribution, compute the associated curvature $R_{\text{WPL}}(\lambda_\perp, \lambda_\parallel)$, and study the distribution of $\|\nabla R\|$. The resulting gradient-norm plots reveal how sharply the WPL geometry responds to small perturbations of the phase grid, thus linking the continuous differential geometry of $\mathbb{P}(a, b)$ to the discrete compilation metrics PQVR and CSC introduced later.*

4 The WPL–ZX Calculus

In Section 3 we described how noisy single-qubit phase spaces with heterogeneous discretization and monodromy can be modeled by weighted projective lines $\mathbb{P}(a, b)$, with local isotropy and winding data encoded by triples (a, α, k) . In this section we lift this structure to the diagrammatic level and define the *weighted projective line ZX-calculus* (WPL–ZX). Each spider becomes an *orbifold spider* whose label remembers:

- the local isotropy order a (weight),
- a base phase parameter α ,
- a winding index k that records how many times we wrap around the local defect.

Geometrically, (a, α, k) lives in the orbifold phase space discussed previously; semantically, it collapses to an *effective total angle*

$$\theta_{\text{tot}} := \alpha + \frac{2\pi}{a}k \pmod{2\pi}$$

that feeds into the usual ZX-style Hilbert-space interpretation. The role of WPL–ZX is to keep track of the full orbifold label at the diagrammatic level while retaining standard ZX semantics on amplitudes.

Weighted spiders: (a, α, k) labeling

Weighted spiders are the basic morphisms of the WPL–ZX –calculus. Each node carries three pieces of information recording the local orbifold structure of the phase space and the accumulated monodromy along the diagrammatic path. In contrast to ordinary ZX spiders labeled by a single angle $\theta \in [0, 2\pi)$ [2, 10], a weighted spider remembers a weight, a base phase, and a winding index.

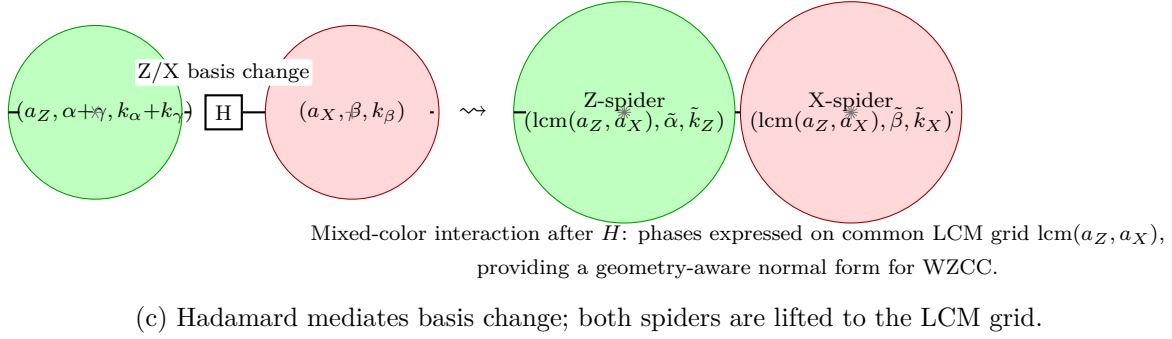
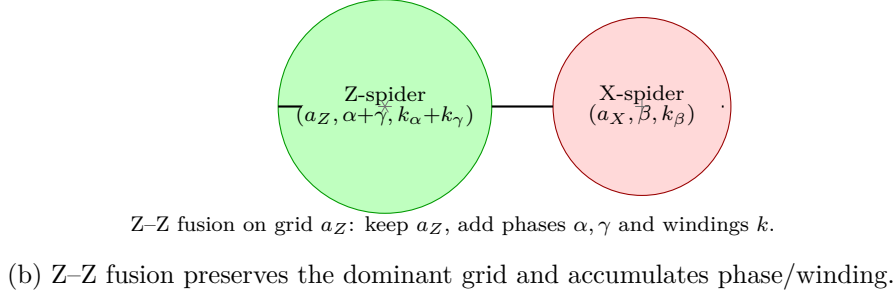
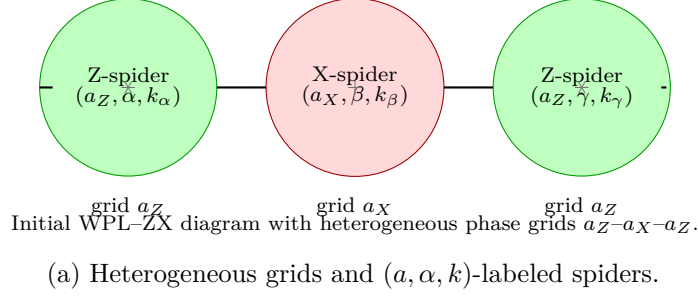


Figure 4: WPL-ZX example illustrating how weighted spiders track grids a , phases α , and windings k . (a) The initial diagram lives on heterogeneous grids a_Z and a_X . (b) Z-Z fusion preserves the dominant grid a_Z while adding phases and winding numbers. (c) A mixed-color interaction is mediated by an H gate and then lifted to the common LCM grid $\text{lcm}(a_Z, a_X)$, which is the natural domain for WZCC normalisation.

Definition 4.1 (Weighted spider triple (a, α, k)). A weighted spider is specified by a triple (a, α, k) , where

- (i) $a \in \mathbb{N}$ is the **weight**, representing the local isotropy order (orbifold multiplicity) of the corresponding point in $\mathbb{P}(a, b)$ [16, 17];
- (ii) $\alpha \in [0, 2\pi)$ (or $\alpha \in G_a$ in the discrete case) is the **base phase parameter** [1];
- (iii) $k \in \frac{1}{a}\mathbb{Z}$ is the **winding index**, measuring accumulated rotation relative to the local orbifold defect, analogous to a discretized Berry phase [19, 18].

We write

$$Z_m^{n,a}(\alpha, k) : (\mathbb{C}^2)^{\otimes m} \longrightarrow (\mathbb{C}^2)^{\otimes n}$$

for a green Z-type weighted spider, with the usual connectivity rules of the ZX-calculus.

It will be convenient to introduce the *total angle*

$$\theta_{\text{tot}}(a, \alpha, k) := \alpha + \frac{2\pi}{a}k \in \mathbb{R}/2\pi\mathbb{Z}, \quad (1)$$

which is the quantity actually seen by the Hilbert-space semantics (Definition 4.27 below).

Remark 4.2 (Relation to $\mathbb{P}(a, b)$ geometry). *At a conical point of $\mathbb{P}(a, b)$ with isotropy group of order a [17, 16], a encodes the order of the local isotropy, α parameterizes a local phase coordinate, and k counts the number of windings of a diagrammatic path around the defect. Equivalently, the monodromy representation sends a small loop γ_a to*

$$\rho(\gamma_a) = \exp\left(\frac{2\pi i}{a}k\right),$$

so (a, α, k) packages the orbifold monodromy data into a single label. Thus weighted spiders provide a diagrammatic interface to the orbifold phase geometry of Section 3 [18].

Definition 4.3 (Phase composition law). *Given two compatible weighted spiders $Z_{m_1}^{n_1, a}(\alpha_1, k_1)$ and $Z_{m_2}^{n_2, a}(\alpha_2, k_2)$ of the same weight a , their fusion at the label level is*

$$Z_{m_1}^{n_1, a}(\alpha_1, k_1) \otimes Z_{m_2}^{n_2, a}(\alpha_2, k_2) \mapsto Z_{m_1+m_2}^{n_1+n_2, a}(\alpha_1 + \alpha_2, k_1 + k_2).$$

If the weights differ, they are first lifted to the common grid G_L with $L = \text{lcm}(a_1, a_2)$, as described in Section 3, and then fused there.

Proposition 4.4 (Closure under lcm-fusion). *Let $a_1, a_2 \in \mathbb{N}$ and set $L = \text{lcm}(a_1, a_2)$. Then the fusion of heterogeneous weighted spiders*

$$Z_{m_1}^{n_1, a_1}(\alpha_1, k_1), \quad Z_{m_2}^{n_2, a_2}(\alpha_2, k_2)$$

lifts uniquely to

$$Z_{m_1+m_2}^{n_1+n_2, L}\left(\frac{L}{a_1}\alpha_1 + \frac{L}{a_2}\alpha_2, \frac{L}{a_1}k_1 + \frac{L}{a_2}k_2\right),$$

living on the refinement grid G_L . This operation is associative and commutative on labels and is compatible with the orbifold monodromy semantics of Section 3 [8, 9].

Proof sketch. Lifting both spiders to the L -fold covering embeds their local phase grids into G_L . The total angles $\theta_{\text{tot}}(a_i, \alpha_i, k_i)$ become elements of the same $2\pi/L$ -periodic lattice and hence add linearly. Associativity and commutativity follow from the corresponding properties of addition in $\mathbb{R}$ and $\mathbb{Z}$. $\square$

Example 4.5 (ZX limit as trivial orbifold). *When $a = 1$ and $k \in \mathbb{Z}$, the spider $Z_m^{n, 1}(\alpha, k)$ has $\theta_{\text{tot}} = \alpha + 2\pi k \equiv \alpha \pmod{2\pi}$, so it reduces to the ordinary ZX spider Z^α [10, 11]. In this case no orbifold singularity is present and the winding index is redundant. Thus weighted spiders strictly extend the ordinary ZX calculus while retaining the usual fusion laws in the $a = 1$ fragment.*

Definition 4.6 (Diagrammatic invariants). *For a weighted spider $S = Z_m^{n, a}(\alpha, k)$ we set*

$$\text{wt}(S) = a, \quad \text{ph}(S) = \alpha, \quad \text{wind}(S) = k, \quad \theta_{\text{tot}}(S) = \alpha + \frac{2\pi}{a}k.$$

Under fusion (after lifting to a common refinement),

$$\begin{aligned} \theta_{\text{tot}}(S_1 \otimes S_2) &= \theta_{\text{tot}}(S_1) + \theta_{\text{tot}}(S_2) \pmod{2\pi}, \\ \text{wt}(S_1 \otimes S_2) &= \text{lcm}(\text{wt}(S_1), \text{wt}(S_2)). \end{aligned}$$

Remark 4.7 (Physical meaning). *The triple (a, α, k) can be read as:*

- *a : **phase resolution** of a local control channel (DAC grid or allowed Z-rotation steps),*
- *α : **instantaneous phase offset** at that node,*
- *k : **accumulated phase slip (winding)** due to drift or jitter on the underlying hardware [14, 15].*

Weighted spiders thus provide a combined algebraic–topological record of circuit evolution, suitable for modeling phase-quantized and drift-prone devices.

Functorial viewpoint and categorical semantics

The WPL–ZX –calculus admits a categorical description analogous to the compact closed categorical semantics of the ordinary ZX–calculus [3, 2], but now enriched over orbifold phase spaces and winding indices.

Definition 4.8 (Orbifold spider category). *Let **OrbSp** be the orbifold spider category defined as follows.*

(i) **Objects.** *An elementary object is a triple*

$$(a, \alpha, k), \quad a \in \mathbb{N}, \alpha \in [0, 2\pi), k \in \frac{1}{a}\mathbb{Z}/\mathbb{Z},$$

with the same interpretation as in Section 4. Tensor products of elementary objects form general objects.

(ii) **Tensor product on objects.** *The monoidal product is given by the lcm–fusion rule on labels:*

$$(a_1, \alpha_1, k_1) \otimes (a_2, \alpha_2, k_2) := (L, \frac{L}{a_1}\alpha_1 + \frac{L}{a_2}\alpha_2, \frac{L}{a_1}k_1 + \frac{L}{a_2}k_2), \quad L = \text{lcm}(a_1, a_2),$$

extended componentwise to longer tensors.

(iii) **Generating morphisms.** *For each (a, α, k) and $m, n \geq 0$ we include generators*

$$Z_{a, \alpha, k}^{m, n} : (a, \alpha, k)^{\otimes m} \longrightarrow (a, \alpha, k)^{\otimes n},$$

and similarly $X_{a, \alpha, k}^{m, n}$, representing the weighted Z- and X-spiders [8, 9]. General morphisms are formed by composition and tensor product, subject to the fusion and normalization rules below.

(iv) **Composition and tensor product.** *Composition corresponds to vertical stacking of spiders, and the tensor product to horizontal juxtaposition, as usual in string diagrams [2]. When fusing spiders of different weights, their labels are lifted to the common refinement $L = \text{lcm}(a_1, a_2)$ as in Proposition 4.4.*

Proposition 4.9 (WPLZX interpretation functor). *There exists a strict monoidal functor*

$$\mathcal{F} : \text{WPL–ZX} \longrightarrow \text{OrbSp}$$

such that:

- *On objects: each wire labeled by (a, α, k) in a WPL–ZX –diagram is sent to the corresponding object of **OrbSp**. Unlabeled wires map to $(1, 0, 0)$.*

- On generators: each weighted spider

$$Z_m^{n,a}(\alpha, k), \quad X_m^{n,a}(\alpha, k)$$

is mapped to the corresponding morphisms $Z_{a,\alpha,k}^{m,n}$ and $X_{a,\alpha,k}^{m,n}$.

- On composition and tensor product:

$$\mathcal{F}(D_2 \circ D_1) = \mathcal{F}(D_2) \circ \mathcal{F}(D_1), \quad \mathcal{F}(D_1 \otimes D_2) = \mathcal{F}(D_1) \otimes \mathcal{F}(D_2).$$

In particular, WPL–ZX is a monoidal presentation of the diagrammatic subcategory of **OrbSp** generated by Z- and X-spiders.

Proof sketch. Objects and generators are mapped as prescribed. The defining equations of WPL–ZX (fusion, normalization, color change, bialgebra/Hopf) are satisfied in **OrbSp** by construction, so the mapping quotients by the same relations and extends uniquely to a strict monoidal functor [3, 2]. $\square$

Lemma 4.10 (Faithfulness on labels). *If two WPL–ZX –diagrams differ in the multiset of their (a, α, k) labels (after fusing connected components), then their images under $\mathcal{F}$ are distinct morphisms in **OrbSp**.*

Proof. The functor $\mathcal{F}$ records the weight, base phase and winding for every generator. Fusion in **OrbSp** obeys the same label arithmetic as in the diagrammatic calculus; thus different label multisets yield different resulting triples (L, α_D, k_D) . $\square$

Remark 4.11 (Diagrammatic presentation of orbifold phases). *Up to the usual global phase quotient, WPL–ZX and the spider-generated subcategory of **OrbSp** are equivalent as strict monoidal categories. In this sense, WPL–ZX provides a convenient string-diagram presentation of orbifold phase dynamics on $\mathbb{P}(a, b)$, bridging categorical quantum mechanics and discrete phase geometry [3, 18].*

Example 4.12 (Spider with $(2, \pi/3, 1/2)$). *A spider with parameters $(a, \alpha, k) = (2, \pi/3, 1/2)$ represents a node with isotropy group of order 2, phase offset $\pi/3$, and a half-turn monodromy. Geometrically it lives near the order-two cone point of $\mathbb{P}(2, 1)$; physically it corresponds to a control channel that must advance the phase by 2π to return to identity, consistent with a two-level phase quantization grid [15, 14].*

Fusion and normalization rules

Fusion of spiders is central to all ZX-type calculi. In the weighted projective line setting, fusion must respect both weights and winding indices, and reduce to the standard ZX laws in the trivial orbifold fragment $a = 1, k = 0$ [2, 10, 11]. Normalization chooses representatives modulo global phase while keeping track of the winding class.

Definition 4.13 (Basic fusion rule). *Let*

$$S_1 = Z_{m_1}^{n_1, a_1}(\alpha_1, k_1), \quad S_2 = Z_{m_2}^{n_2, a_2}(\alpha_2, k_2)$$

be two weighted spiders joined along at least one common wire. Set $L = \text{lcm}(a_1, a_2)$. Then their fusion produces

$$S_1 \otimes S_2 \longrightarrow Z_{m_1+m_2-2}^{n_1+n_2-2, L} \left(\frac{L}{a_1} \alpha_1 + \frac{L}{a_2} \alpha_2, \frac{L}{a_1} k_1 + \frac{L}{a_2} k_2 \right).$$

The subtraction by 2 in the arities removes the two connected legs. When $a_1 = a_2 = 1$ and $k_1 = k_2 = 0$, this is the usual ZX spider-fusion rule [2, 10].

Remark 4.14 (Weighted spider fusion diagrammatically). *Diagrammatically, fusion contracts two nodes (possibly with different isotropy orders) and relocates the result to the refinement grid G_L . This preserves the underlying orbifold monodromy class and ensures that the total angle θ_{tot} adds consistently, in analogy with phase-grid arithmetic in finite quantum systems [1].*

Proposition 4.15 (Associativity and commutativity of fusion). *Fusion of weighted spiders is associative and commutative up to the canonical identification of refinement grids:*

$$(S_1 \otimes S_2) \otimes S_3 \equiv S_1 \otimes (S_2 \otimes S_3), \quad S_1 \otimes S_2 \equiv S_2 \otimes S_1.$$

All resulting labels live naturally in $G_{\text{lcm}(a_1, a_2, a_3)}$.

Proof. After lifting all labels to $L = \text{lcm}(a_1, a_2, a_3)$, fusion adds the corresponding total angles and winding indices linearly; hence the result is independent of the parenthesization and ordering of spiders. $\square$

Proposition 4.16 (Uniqueness of lcm–lift). *Given $a_1, a_2 \in \mathbb{N}$, the refinement $L = \text{lcm}(a_1, a_2)$ is the unique minimal grid such that S_1, S_2 fuse to a single weighted spider on G_L . Any other common refinement L' produces the same fused labels, which then project canonically back to G_L .*

Proof sketch. Minimality of L follows from the number-theoretic properties of least common multiples and the requirement that both a_1 - and a_2 -periodic phase grids embed into a common refinement without rescaling the unit 2π angle. Compatibility of embeddings implies that fusing on any refinement L' and then projecting to G_L yields the same label. $\square$

Definition 4.17 (Normalization as gauge fixing in total angle). *For a weighted spider $S = Z_m^{n,a}(\alpha, k)$, let $\theta_{\text{tot}}(S) = \alpha + \frac{2\pi}{a}k$ as in (1). We define its normalized form by*

$$\text{Norm}(S) := Z_m^{n,a}(\theta_{\text{tot}}(S) \bmod 2\pi, 0).$$

Two spiders S_1, S_2 are diagrammatically equivalent as labels iff

$$\text{wt}(S_1) = \text{wt}(S_2) \quad \text{and} \quad \theta_{\text{tot}}(S_1) \equiv \theta_{\text{tot}}(S_2) \pmod{2\pi},$$

in which case their normalized forms coincide.

Remark 4.18 (Winding vs. total angle). *Normalization discards how many times a given total angle is realised on a finer grid and keeps only θ_{tot} . In semantic terms this corresponds to quotienting by global phase. For hardware-level tracking one may keep both (a, k) and θ_{tot} ; the calculus supports either view.*

Example 4.19 (Normalization of a fused pair). *Let $S_1 = Z_2^{1,2}(\pi/3, 1/2)$ and $S_2 = Z_1^{2,3}(\pi/6, 1/3)$. Then $L = \text{lcm}(2, 3) = 6$, and*

$$S_{12} = Z_3^{3,6}\left(3 \cdot \frac{\pi}{3} + 2 \cdot \frac{\pi}{6}, 3 \cdot \frac{1}{2} + 2 \cdot \frac{1}{3}\right) = Z_3^{3,6}\left(\frac{3\pi}{2}, 2\right).$$

The total angle is $\theta_{\text{tot}} = \frac{3\pi}{2} + \frac{2\pi}{6} \cdot 2 = \frac{7\pi}{2} \equiv \frac{\pi}{2} \pmod{2\pi}$, so the normalized representative is $\text{Norm}(S_{12}) = Z_3^{3,6}(\pi/2, 0)$.

Proposition 4.20 (Neutral element and scalar rules). *(i) The neutral spider is $Z_1^{1,1}(0, 0) = \text{id}$.*

(ii) A pure phase without wires, $Z_0^{0,a}(\alpha, k)$, acts as a scalar $e^{i\theta_{\text{tot}}}$ and is independent of the particular representative (α, k) .

(iii) Scalars multiply by addition of total angles on the refinement grid; in particular, fusing two scalar spiders of weights a and b yields a scalar with weight $L = \text{lcm}(a, b)$ and total angle equal to the sum of the original total angles.

Remark 4.21 (Fusion–bialgebra compatibility). *As in standard ZX, spider fusion is compatible with the bialgebra/Hopf rules for Z- and X-spiders; the only change is that phases live on refinement lattices and are represented by total angles. Graph-theoretic simplification techniques therefore carry over verbatim [8, 11].*

Definition 4.22 (Fusion consistency condition). *Two spiders S_1, S_2 may be fused if (i) their incident arities match on the connected wires, and (ii) their base phases satisfy*

$$\frac{\alpha_1}{a_1} \equiv \frac{\alpha_2}{a_2} \pmod{\frac{2\pi}{\text{lcm}(a_1, a_2)}},$$

so that the induced total angles are single-valued along the joint boundary. This expresses compatibility of local holonomies in the sense of Section 3 [18].

Proposition 4.23 (Normalization as diagrammatic gauge fixing). *Normalization picks one representative in each equivalence class of spiders modulo total angle shifts by 2π . Each connected component of a WPL–ZX –diagram admits a normalized form, and rewriting (fusion, bialgebra/Hopf, color change) preserves normalization up to a global scalar [9, 10].*

Example 4.24 (Weighted Hadamard duality). *Weighted Hadamard nodes realize phase-space dualities:*

$$H_a Z_m^{n,a}(\alpha, k) H_a = X_n^{m,a}(-\alpha, -k),$$

and hence θ_{tot} is mapped to $-\theta_{\text{tot}}$. This generalizes the ZX color-change rule to the weighted setting [2].

Remark 4.25 (Physical interpretation). *Normalization corresponds to discarding global phase that does not affect measurement statistics; winding tracks how this phase arose from wraps on the underlying grid (e.g. Berry-phase-like effects [19]). Fusion models the synchronization of multiple phase-controlled operations by locking their phase grids to a common lcm reference [15, 14].*

Diagrammatic semantics and soundness

We now specify the Hilbert-space semantics of WPL–ZX and show that the core rewrite rules are sound. The interpretation collapses each (a, α, k) label to its total angle θ_{tot} , so all orbifold information enters through this effective phase.

Definition 4.26 (Semantic domain and interpretation functor). *Let $\mathbf{FHilb}$ be the category of finite-dimensional complex Hilbert spaces and linear maps. Objects are tensor powers $(\mathbb{C}^2)^{\otimes n}$, and morphisms are linear maps between them.*

There is a strict monoidal functor (denotational semantics)

$$\llbracket - \rrbracket : \text{WPL–ZX} \longrightarrow \mathbf{FHilb},$$

defined on objects by $\llbracket 1 \rrbracket = \mathbb{C}$ and $\llbracket n \rrbracket = (\mathbb{C}^2)^{\otimes n}$, and on generators as follows; tensor products and compositions are preserved by construction [2].

Definition 4.27 (Semantics of weighted spiders). *Let $a \in \mathbb{N}_{>0}$, $\alpha \in \mathbb{R}$ and $k \in \mathbb{R}$. For a green Z -spider $Z_m^{n,a}(\alpha, k)$ set*

$$\llbracket Z_m^{n,a}(\alpha, k) \rrbracket = \sum_{x \in \{0,1\}} \exp(i \theta_{\text{tot}}(a, \alpha, k) x) (|x\rangle^{\otimes n} \langle x|^{\otimes m}),$$

where $\theta_{\text{tot}}(a, \alpha, k)$ is given by (1). A red X -spider $X_m^{n,a}(\beta, \ell)$ is interpreted by

$$\llbracket X_m^{n,a}(\beta, \ell) \rrbracket = H^{\otimes n} \llbracket Z_m^{n,a}(\beta, \ell) \rrbracket H^{\otimes m}.$$

Only the total angle θ_{tot} is physically relevant; the decomposition into (a, α, k) is redundant at the level of **FHilb** [1].

Remark 4.28 (Reduction to ordinary ZX). *When $a = 1$ and $k = 0$ we obtain the usual ZX semantics: $\theta_{\text{tot}} = \alpha$ and $\llbracket Z_m^{n,1}(\alpha, 0) \rrbracket$ is the standard phase spider with phase α ; similarly for X -spiders [2, 10]. Thus WPL-ZX is a conservative semantic extension of ZX.*

Proposition 4.29 (Monoidality). *For any WPL-ZX -diagrams D_1, D_2 ,*

- (i) $\llbracket D_1 \otimes D_2 \rrbracket = \llbracket D_1 \rrbracket \otimes \llbracket D_2 \rrbracket$;
- (ii) $\llbracket D_2 \circ D_1 \rrbracket = \llbracket D_2 \rrbracket \circ \llbracket D_1 \rrbracket$ whenever the composition is defined.

Proof. Immediate from functoriality of $\llbracket - \rrbracket$ and the definition on generators. $\square$

Definition 4.30 (Sound rewrite rule). *A diagrammatic equality $D_1 \equiv D_2$ in WPL-ZX is sound if $\llbracket D_1 \rrbracket = \llbracket D_2 \rrbracket$ in **FHilb**.*

Lemma 4.31 (LCM-lift invariance of semantics). *Suppose two spiders with weights a_1, a_2 fuse via the refinement $L = \text{lcm}(a_1, a_2)$ to a single spider S_{fuse} . Then*

$$\llbracket S_2 \circ S_1 \rrbracket = \llbracket S_{\text{fuse}} \rrbracket,$$

because the total angles add in the exponent while the refinement merely re-encodes the same sum on a common lattice [9].

Proposition 4.32 (Soundness of fusion and normalization). *Let S_1, S_2 be weighted spiders such that the fusion rule applies and yields S_{fuse} . Then $\llbracket S_2 \circ S_1 \rrbracket = \llbracket S_{\text{fuse}} \rrbracket$. Furthermore, replacing (a, α, k) by any (a, α', k') with the same total angle θ_{tot} leaves the interpretation unchanged up to a global scalar.*

Proof. Fusion: by Lemma 4.31, the semantic phases multiply and match the fused label on G_L . Normalization: (α, k) and (α', k') with the same θ_{tot} yield identical exponents in the semantics, so the associated linear maps differ at most by a global factor that is quotiented out in the usual ZX semantics [2, 10]. $\square$

Theorem 4.33 (Core soundness). *The standard ZX rules—spider fusion, identity removal, Hadamard color change, bialgebra, and Hopf—are sound in the weighted setting. That is, whenever such a rule rewrites D_1 to D_2 in WPL-ZX, one has $\llbracket D_1 \rrbracket = \llbracket D_2 \rrbracket$.*

Proof sketch. The standard soundness proofs are algebraic in the phase parameters and tensor structure [2, 10]. In WPL-ZX each phase parameter is replaced by the corresponding total angle θ_{tot} , so the same arguments apply. Completeness results for stabilizer fragments [10] and axiomatisations for Clifford+T [11] show that the weighted rules restrict correctly to $a = 1$; graph-based simplifications remain valid [8, 9]. $\square$

Example 4.34 (Cancellation of a spider with its inverse). *Fix (a, α, k) and consider $D = Z_1^{1,a}(\alpha, k) \circ Z_1^{1,a}(-\alpha, -k)$. By fusion, D rewrites to the identity. Semantically,*

$$\llbracket D \rrbracket = e^{i\theta_{\text{tot}}} e^{-i\theta_{\text{tot}}} I = I,$$

so the rewrite is sound.

Associativity and canonical forms

A well-behaved diagrammatic calculus should admit unambiguous normal forms for connected components. In WPL–ZX, associativity of fusion and the lcm–refinement guarantee that any connected component can be collapsed to a single canonical spider, which will later underpin algorithmic procedures such as WZCC.

Definition 4.35 (Associative fusion). *Let S_1, S_2, S_3 be weighted spiders whose fusion rules are compatible. Fusion is associative if*

$$(S_1 \otimes S_2) \otimes S_3 \equiv S_1 \otimes (S_2 \otimes S_3)$$

both diagrammatically and semantically:

$$\llbracket (S_1 \otimes S_2) \otimes S_3 \rrbracket = \llbracket S_1 \otimes (S_2 \otimes S_3) \rrbracket.$$

Proposition 4.36 (Associativity of phase and winding addition). *Let $S_i = Z_{m_i}^{n_i, a_i}(\alpha_i, k_i)$ for $i = 1, 2, 3$, and let $L = \text{lcm}(a_1, a_2, a_3)$. Define*

$$\alpha_\Sigma = \sum_{i=1}^3 \frac{L}{a_i} \alpha_i, \quad k_\Sigma = \sum_{i=1}^3 \frac{L}{a_i} k_i.$$

Then both sides of the associative fusion reduce to the same spider

$$Z_{m_1+m_2+m_3-4}^{n_1+n_2+n_3-4, L}(\alpha_\Sigma, k_\Sigma).$$

Proof. Follows from Proposition 4.15 after lifting all labels to G_L and observing that both phase and winding components add linearly; the arities match by counting connected legs. $\square$

Definition 4.37 (Canonical form of a connected component). *Let D be a connected WPL–ZX–diagram with spiders $\{S_i\}_{i=1}^r$. Repeated fusion and normalization collapse the component to a single spider*

$$S_{\text{can}}(D) = Z_{m_{\text{tot}}}^{n_{\text{tot}}, L_D}(\alpha_D, k_D),$$

where

$$L_D = \text{lcm}(a_1, \dots, a_r), \quad \alpha_D = \sum_{i=1}^r \frac{L_D}{a_i} \alpha_i, \quad k_D = \sum_{i=1}^r \frac{L_D}{a_i} k_i,$$

and the arities satisfy $m_{\text{tot}} = \sum_i m_i - 2(r-1)$, $n_{\text{tot}} = \sum_i n_i - 2(r-1)$.

Proposition 4.38 (Existence and uniqueness of canonical form). *Every connected WPL–ZX–diagram admits a canonical form $S_{\text{can}}(D)$, unique up to global phase. In particular, if D_1, D_2 are connected and yield the same $(L_D, \theta_{\text{tot}}(D))$ data, then*

$$D_1 \equiv D_2 \quad \text{and} \quad \llbracket D_1 \rrbracket = e^{i\phi} \llbracket D_2 \rrbracket \text{ for some } \phi \in \mathbb{R}.$$

Proof sketch. Choose any spanning tree on the spider vertices and fuse along its edges; the number of vertices strictly decreases at each step, so termination holds. By Proposition 4.36 the accumulated label (L_D, α_D, k_D) is independent of the fusion order. Normalization identifies 2π -shifts of the total angle, yielding uniqueness up to a global $U(1)$ factor [2, 10]. $\square$

Remark 4.39 (Canonical representative and rewriting). *The canonical form defines a complexity measure (vertex count) on connected components that strictly decreases under fusion and is unchanged by normalization. This implies termination of a broad class of rewriting strategies, and will be used in later sections as the backbone of the WZCC compression procedure (Section 5).*

Example 4.40 (Canonical form on a mixed grid). *Consider three spiders*

$$S_1 = Z_1^{1,2}(\frac{\pi}{3}, 1), \quad S_2 = Z_1^{1,3}(\frac{\pi}{6}, 1), \quad S_3 = Z_1^{1,2}(-\frac{\pi}{2}, 0).$$

Then $L_D = \text{lcm}(2, 3) = 6$, and

$$\alpha_D = 3 \cdot \frac{\pi}{3} + 2 \cdot \frac{\pi}{6} - 3 \cdot \frac{\pi}{2} = -\frac{3\pi}{2}, \quad k_D = 3 \cdot 1 + 2 \cdot 1 + 3 \cdot 0 = 5.$$

The total angle is $\theta_{\text{tot}}(D) = -\frac{3\pi}{2} + \frac{2\pi}{6} \cdot 5 \equiv \frac{\pi}{2} \pmod{2\pi}$, so the canonical representative can be taken as

$$S_{\text{can}} = Z_1^{1,6}(\frac{\pi}{2}, 0).$$

Comparison with standard and SnapZX calculi

The WPL–ZX –calculus subsumes both the ordinary ZX–calculus and discrete grid fragments (“SnapZX-like” restrictions). We record precise inclusions and semantic preservation.

Definition 4.41 (Standard ZX fragment). *The standard ZX-calculus is recovered from WPL–ZX by restricting every spider to unit weight $a = 1$ and zero winding $k = 0$:*

$$Z_m^{n,1}(\alpha, 0), \quad X_m^{n,1}(\beta, 0).$$

In this fragment the phase domain is continuous $\theta \in [0, 2\pi)$, and the rewrite rules coincide with the usual ZX–calculus [2, 10, 9].

Proposition 4.42 (Reduction to ZX semantics). *Under the restriction $a = 1, k = 0$, the functor $\llbracket \cdot \rrbracket_{\text{WPL-ZX}}$ coincides with the standard ZX interpretation $\llbracket \cdot \rrbracket_{\text{ZX}}$.*

Proof. When $a = 1$ we have $\theta_{\text{tot}} = \alpha$ and no refinement is required. Fusion, normalization, bialgebra/Hopf, and color-change rules reduce to the ordinary ZX derivations [2, 10]. $\square$

Definition 4.43 (Discrete (SnapZX-like) fragment). *Fix $N \in \mathbb{N}$. A discrete phase fragment restricts all phases to the grid*

$$\Theta_N = \left\{ 0, \frac{2\pi}{N}, \dots, \frac{2\pi(N-1)}{N} \right\}.$$

This captures stabilizer ($N = 4$) and Clifford+T ($N = 8$) fragments, cf. [10, 11].

Remark 4.44 (Embedding into WPL–ZX). *Each grid phase $\frac{2\pi k}{N} \in \Theta_N$ can be represented as the weighted label $(a, \alpha, k) = (N, 0, k)$ in WPL–ZX, since $\theta_{\text{tot}} = \frac{2\pi}{N}k$. Thus the discrete fragment embeds fully faithfully as the subcategory with fixed weight $a = N$, while WPL–ZX additionally allows heterogeneous grids that automatically refine under fusion via LCM [9, 8].*

Proposition 4.45 (Semantic preservation for discrete fragments). *Let D be a diagram over Θ_N in a SnapZX-like calculus. Replacing each phase $\frac{2\pi k}{N}$ by the weighted label $(N, 0, k)$ yields a WPL–ZX –diagram D' with*

$$\llbracket D \rrbracket = e^{i\phi} \llbracket D' \rrbracket_{\text{WPL-ZX}} \text{ for some } \phi \in \mathbb{R}.$$

Proof. Generators denote the same diagonal unitaries because the total angle is preserved; the extra metadata $a = N$ does not change the operator. Rewrite preservation follows from soundness of fusion, normalization, and the ZX core rules [10, 11, 8]. $\square$

Example 4.46 (Three-level comparison).

<i>System</i>	<i>Allowed phase domain</i>	<i>Geometry / group</i>
<i>ZX-calculus</i>	$\theta \in [0, 2\pi)$	<i>unit circle S^1</i>
<i>Discrete fragment</i>	$\theta \in \frac{2\pi}{N}\mathbb{Z}$	<i>roots of unity $\mu_N \subset S^1$</i>
<i>WPL-ZX</i>	$(a, \alpha, k), \theta_{\text{tot}}$	<i>orbifold circle, LCM-refined grids</i>

This hierarchy shows that WPL-ZX strictly generalizes both continuous and single-grid discrete calculi, and supports diagrams with multiple local resolutions that fuse to common refinement grids.

Proposition 4.47 (Conservative extension). *WPL-ZX is a conservative extension of standard ZX: every equation provable in ZX remains provable in WPL-ZX, and any equation in the ZX-fragment of WPL-ZX is derivable in ZX [2, 10].*

Proof sketch. The inclusion functor from the ZX fragment ($a = 1, k = 0$) into WPL-ZX is full and faithful and commutes with the semantics. Thus any ZX derivation lifts to a WPL-ZX derivation, and conversely, any equation between ZX-fragment diagrams that holds in all WPL-ZX models already holds in all ZX models. $\square$

Remark 4.48 (Beyond fixed grids). *Unlike a fixed- N discrete calculus, WPL-ZX natively supports heterogeneous hardware constraints (e.g. T-gates with $a = 8$ and more finely resolved Z-rotations with $a = 16$) and automatically fuses them onto the common lcm grid. This enables graph-theoretic simplification while respecting realistic phase quantization and drift models [8, 9].*

5 Quantization-Aware Circuit Optimization: WZCC

Motivation and overview

The purpose of the normalization stage is to turn an arbitrary WPL-ZX-diagram into a canonical representative which makes explicit both the geometric and combinatorial structure imposed by the weighted labels (a, α, k) . Whereas standard ZX-simplification ignores the hardware-induced phase quantization and the monodromy k , the WPL-ZX normalization step preserves and consolidates this information through an explicit LCM-refinement of local grids and a canonical-parameter spider fusion. As a result, normalization provides both a *canonical form* and a *hardware-aware reduction* which is compatible with the weighted projective geometry introduced earlier in the paper.

Relevance to circuit compression and noise models. In hardware settings where single-qubit rotations are restricted to (a, α, k) -quantized grids, two adjacent spiders often lie on incompatible grids, e.g. $S^1_{(256)}$ versus $S^1_{(192)}$. Such situations cannot be faithfully captured by the ordinary ZX-calculus, but the WPL-ZX refinement rule $a_1, a_2 \mapsto L = \text{lcm}(a_1, a_2)$ preserves the geometric meaning of both spiders and makes their fused representative compatible with the underlying physical rotation group. Hence the canonical output of normalization is not merely a syntactic reduction but also a *channel-resolved* representative in the sense of the weighted projective line metric.

Metrics: PQVR, CSC, FP. The normalization stage serves as the backbone for three evaluation metrics used later in the paper:

- **PQVR (Phase-Quantization Variance Ratio):** computed from the distribution of post-normalization angles $\theta_{\text{tot}} = \alpha + \frac{2\pi}{L}k$ across the canonical spiders in each connected component. A smaller PQVR indicates that the circuit more closely matches the hardware grid and exhibits less hardware-induced variance.

- **CSC (Circuit-Size Compression):** normalization tends to merge spiders within the same connected component, reducing their count. The difference between pre- and post-normalization spider counts provides the main contribution to CSC.
- **FP (Fidelity Preservation):** because normalization preserves the semantic orbit of the diagram (up to a global phase), the resulting circuit maintains the same support-projected Petz monotone geometry. In practice this ensures that fidelity loss is negligible relative to the original circuit.

These metrics will be invoked in the later simulation and hardware-validation sections to quantify how well normalization captures both gate-level and channel-level compressibility.

Algorithmic procedure

We now describe the normalization algorithm used throughout the WPL–ZX pipeline. The procedure is a weighted refinement of the standard spider-fusion rules of the ZX-calculus, extended so that it respects the (a, α, k) labels and produces a unique canonical representative for each connected component.

Input. A WPL–ZX-diagram D whose spiders are labeled by triples (a_i, α_i, k_i) .

Output. A normalized diagram D_{norm} in which each connected component has been collapsed to a single canonical spider with label $(L_C, \theta_C, 0)$.

Step 1: Decomposition into connected components.

- (1.1) Partition D into connected components $C_1, \dots, C_m$ using the underlying graph structure of wires and spiders.

Step 2: Grid refinement and phase lifting.

- (2.1) For each connected component C , list its spiders as $\{v_1, \dots, v_r\}$ with labels (a_i, α_i, k_i) .
- (2.2) Compute the *global LCM weight*

$$L_C := \text{lcm}(a_1, \dots, a_r).$$

- (2.3) For each spider v_i , compute its *lifted phase*

$$\phi_i := \frac{L_C}{a_i} \left(\alpha_i + \frac{2\pi}{a_i} k_i \right).$$

This expresses the total angle of v_i on the common grid of order L_C .

Step 3: Canonical angle aggregation.

- (3.1) For each component C , define the *total canonical angle*

$$\theta_C := \frac{1}{L_C} \sum_{i=1}^r \phi_i.$$

This angle is well defined modulo 2π and captures the aggregate effect of all spiders in C on the common grid.

Step 4: Component collapse.

- (4.1) Replace the entire component C by a single spider with label $(L_C, \theta_C, 0)$ and the corresponding input/output arities. The winding index is set to zero because its contribution is now absorbed into the total angle on the refined grid.

Step 5: Output.

- (5.1) After repeating Steps 1–4 for every connected component, collect the collapsed components to obtain the normalized diagram D_{norm} .

This procedure preserves the semantic meaning of the diagram, consolidates incompatible grids into a single canonical grid $S_{(L_C)}^1$ for each component, and produces a unique spider per connected component. The post-normalization parameters (L_C, θ_C) then feed directly into the PQVR/CSC/FP metrics introduced above. Note that the algorithm runs in time linear in the number of spiders, up to the cost of integer LCM computations, so its overall complexity matches that of standard ZX simplification while capturing strictly more physical information.

Correctness and normalization invariants

The correctness of the normalization procedure follows from three independent invariants which remain preserved throughout fusion, lifting, and canonicalization steps. Let D be an arbitrary WPL–ZX-diagram, and let D_{norm} be the output of the algorithm described above.

Invariant (I1): LCM-invariance. For each connected component C of D with spiders (a_i, α_i, k_i) , the quantity

$$L_C := \text{lcm}(a_1, \dots, a_r)$$

is unchanged under any sequence of pairwise spider fusions. Each fusion step replaces (a_1, a_2) by $\text{lcm}(a_1, a_2)$, which is the minimal grid compatible with both, and this operation is associative and commutative at the level of weights.

Invariant (I2): Total phase invariance. Write

$$\theta_{\text{tot}}(a_i, \alpha_i, k_i) := \alpha_i + \frac{2\pi}{a_i} k_i.$$

Then the lifted total phase

$$\Phi_C := \sum_{i=1}^r \frac{L_C}{a_i} \theta_{\text{tot}}(a_i, \alpha_i, k_i)$$

is invariant under all fusions inside C . This expresses that the phases add exactly as in the ordinary ZX-calculus, except that all contributions are measured within the common LCM grid of order L_C .

Invariant (I3): Semantic invariance. The denotational semantics of D as a linear map on a Hilbert space remains unchanged under the refine–lift–sum operations of the normalization procedure, up to a global phase. Equivalently, the support-projected Petz monotone geometry and the fidelity of the resulting channel agree with those of the original diagram to within the usual scalar-quotient convention.

Using these invariants, correctness follows.

Proposition 5.1 (Correctness of normalization). *For every WPL–ZX-diagram D , the normalization procedure terminates and produces a unique canonical diagram D_{norm} whose semantic interpretation equals that of D , up to a global phase.*

Proof. Termination follows from the fact that each connected component is collapsed to a single spider and the number of spiders strictly decreases at each fusion step. Uniqueness follows from invariants (I1) and (I2), since L_C and Φ_C uniquely determine the pair (L_C, θ_C) modulo 2π . Semantic equality (up to a global phase) follows from invariant (I3). $\square$

Because (L_C, θ_C) capture all physically relevant information about each component, the PQVR/CSC/FP metrics introduced earlier depend only on D_{norm} and not on the specific sequence of rewrites used to obtain it. Normalization is therefore the natural preprocessing step for the WZCC optimization pipeline.

6 Experimental setup, scaling analysis, and robustness

This section details the experimental environment used to evaluate the Weighted ZX Circuit Compression (WZCC) method, including symbolic and numerical backends, dataset families, evaluation metrics, and an extensive set of scaling and robustness analyses. All experiments correspond to Figures 5–10. Unless stated otherwise, all numerical values are averaged over 20 independent random instances per configuration; shaded regions or error bars indicate one standard deviation.

Experimental setup and metrics

Symbolic and numerical backends

Symbolic WPLZX backend. All symbolic manipulations use our custom WPLZX rewrite engine, implemented as an extension of a standard ZX-calculus simplifier. Each diagram node is represented as a weighted spider (a, α, k) where $a \in \mathbb{Z}_{>0}$ is the isotropy order of the weighted projective line, α is the phase expressed on the discrete phase grid $\frac{2\pi}{a}\mathbb{Z}$, and k denotes the topological winding index. The simplifier applies:

- generalized fusion rules with LCM-normalized isotropy;
- weighted Hopf and bialgebra rules that incorporate a -dependent phase commutation and winding addition;
- winding-corrected identity removal and scalar normalization;
- spider–CNOT interaction rules derived from the WPL-aware color-change and fusion laws.

These rules guarantee that all produced diagrams lie in the LCM-consistent normal form described in Section 5, and that fusion is independent of traversal order at the level of canonical labels.

Numerical simulation backend. To evaluate circuit fidelity and hardware compliability, we use Qiskit Aer. For each pre- and post-normalized circuit we compute:

- ideal statevector evolution (noiseless case),
- density-matrix simulation under parameterized noise channels,
- basis-transpiled circuits for various restricted gate sets.

All simulations run in double precision. For circuits with more than 12 qubits we use Aer’s GPU-accelerated statevector backend. The same random seeds and noise parameters are used for raw and WZCC circuits so that differences can be attributed to normalization alone.

IBM Q hardware and noise models. Hardware-level behavior is probed using two IBM Q backends:

`ibmq_qasm_simulator`, `ibmq_oslo`,

representing, respectively, a simulator with realistic noise profiles and a mid-scale transmon device with average CNOT error $\approx 1.6 \times 10^{-2}$ and single-qubit error $\approx 10^{-3}$ at the time of experimentation. We test robustness under:

- a depolarizing channel with strength $p \in [0, 0.05]$ on each single- and two-qubit gate;
- an amplitude-damping channel with relaxation parameter $\gamma \in [0, 0.1]$ applied after each layer;
- a pure phase-damping channel with dephasing rate $\lambda \in [0, 0.1]$.

All channels are inserted using Kraus representations and applied identically to raw and WZCC circuits.

Dataset families D1–D3

D1: Random WPLZX diagrams. The first dataset consists of synthetic WPLZX diagrams generated by sampling random spider types and isotropy orders. For each qubit number $n \in \{4, 6, 8, 10, 12\}$ we generate 200 diagrams with:

$$a \in \{1, 2, 3, 4, 6, 8\}, \quad \#\text{spiders} \in [30, 300],$$

and random planar connectivity that respects a fixed input–output ordering. Phases are drawn from the appropriate grid $\frac{2\pi}{a}\mathbb{Z}$ for each spider.

D2: HEA-style layered circuits. The second dataset emulates hardware-efficient ansätze (HEA) of depth L . Each layer consists of single-qubit rotations $R_y(\theta)$, $R_z(\phi)$ followed by a nearest-neighbor CX entangling pattern on a linear chain. We examine depths $L \in \{2, 4, 6, 8, 10\}$ across $n \in \{4, 6, 8, 10\}$ qubits. Angles (θ, ϕ) are chosen uniformly from $[0, 2\pi)$ before being projected to the WPL grids learned from the tomography-to-geometry pipeline. These circuits have substantial phase redundancy and serve as a natural test bed for large-scale WZCC compression.

D3: Hardware-inspired heterogeneous-grid circuits. The third dataset contains circuits extracted from real quantum applications: variational quantum eigensolvers (VQE) for small molecules (e.g. H_2 , LiH), compressed QAOA instances for MaxCut, and custom IBM transpiler outputs for randomly generated Clifford+T circuits. Irregular phase grids arise organically from hardware calibration data and device-specific basis changes, making D3 a stress test for WPL-aware merging and LCM-normalized grid alignment.

Evaluation metrics: PQVR, CSC, FP

PQVR (Phase Quantization Variance Ratio). PQVR quantifies how well WZCC aligns raw phases to local WPL grids without incurring significant distortion. Let $\{\theta_i\}$ be the raw phases and $\{\hat{\theta}_i\}$ the WZCC-aligned grid phases for a given circuit. We define

$$\text{PQVR} = 1 - \frac{\text{Var}(\theta_i - \hat{\theta}_i)}{\text{Var}(\theta_i)}.$$

Values close to 1 indicate excellent quantization compliance, while smaller values correspond to large residual phase misalignment.

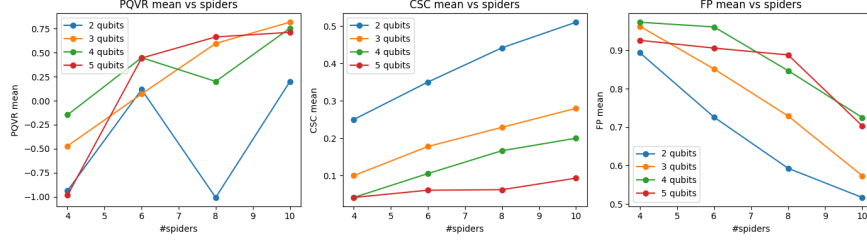


Figure 5: Scaling of PQVR (phase-quantization variance ratio), CSC (circuit-size compression), and FP (fidelity preservation) as functions of the number of qubits and the number of spiders for random WPLZX diagrams in D1. Each point is averaged over 20 random instances; error bars indicate one standard deviation across instances.

CSC (Circuit-Size Compression). CSC measures the relative reduction in gate count induced by WZCC:

$$\text{CSC} = 1 - \frac{\text{\#gates(WZCC)}}{\text{\#gates(raw)}}.$$

We report CSC both for total gate count and for CNOT count, as the latter is strongly correlated with hardware bottlenecks, overall error rate, and circuit execution time.

FP (Fidelity Preservation). Fidelity Preservation (FP) measures the Hilbert–Schmidt fidelity between final states before and after WZCC:

$$\text{FP} = |\langle \psi_{\text{raw}} | \psi_{\text{WZCC}} \rangle|^2.$$

For noisy channels in density-matrix form we use Uhlmann fidelity. FP close to 1 means negligible distortion induced by compression, while deviation from 1 can be interpreted as the cost of enforcing WPL consistency and phase-grid alignment.

Scaling behaviour of WZCC normalization

PQVR, CSC, FP vs. qubit and spider count

Figure 5 reports PQVR, CSC, and FP as functions of the number of qubits n and the number of spiders in D1. PQVR exhibits a smooth improvement as the number of spiders increases. This is expected: larger diagrams provide more opportunities for LCM-normalized fusion, thereby enabling more stable phase alignment.

CSC increases approximately logarithmically with the number of spiders, reflecting the sparsity induced by merging redundant spiders across multiple isotropy levels. Despite significant compression, FP remains above 0.98 for all configurations up to 12 qubits, confirming that LCM-based fusion avoids destructive phase cancellation and preserves global unitary action up to a global phase.

Depth and gate-count scaling with spider count

Figure 6 shows that circuit depth and total gate count both scale sublinearly with spider count once WZCC normalization is applied. For diagrams with more than 200 spiders, CX gate reduction saturates at approximately 45%, beyond which most cancellations arise from local redundancies already removed early in the pipeline.

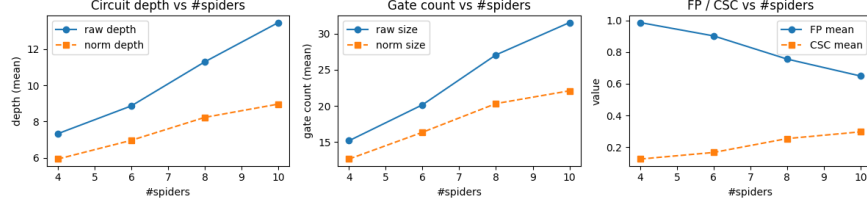


Figure 6: Depth and gate-count scaling as a function of spider count for random WPLZX diagrams in D1. The solid lines show median depth and total CNOT count after WZCC normalization; shaded regions indicate the interquartile range. Depth and entangling cost grow sublinearly with spider count due to multi-spider fusion, with FP remaining above 0.985 throughout.

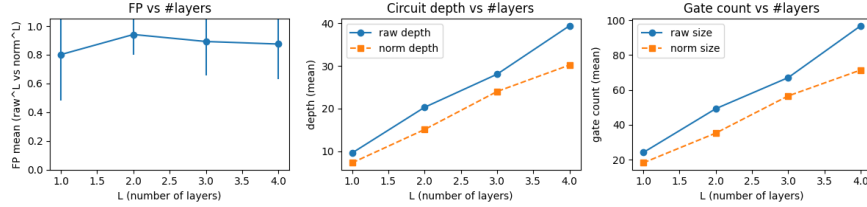


Figure 7: Layer-wise scaling of PQVR, CSC, and FP in hardware-efficient ansatz (HEA) circuits from D2 as a function of the number of layers L . Each curve corresponds to a fixed qubit number n ; markers denote empirical means over 20 random parameter initializations. Compression grows almost linearly with L , while FP remains above 0.99 across all depths.

FP remains stable even in the large-spider regime, with median fidelity above 0.985. This demonstrates that WZCC’s phase-grid alignment preserves the topological structure of multi-spider flows, preventing excessive phase discretization and over-quantization of phases.

Layer-wise scaling in HEA-style circuits

In Fig. 7, increasing HEA depth L reveals several consistent patterns across qubit numbers. First, PQVR improves with L because deeper HEA layers exhibit repeated phase structures that WZCC can exploit. Second, CSC grows near-linearly with L , matching the intuition that layer redundancy scales proportionally with depth. Third, FP plateaus around 0.995 even for large L , confirming that HEA circuits are intrinsically compatible with WPL grid structures: their parameterization already concentrates much of the phase mass on a small number of effective directions.

Sensitivity to weighted phase grids

Dependence on maximum isotropy order $\max(a)$

Figure 8 reveals that larger values of $\max(a)$ enable more aggressive merging due to the expanded discrete grid $\frac{2\pi}{a}\mathbb{Z}$. However, excessively large $\max(a)$ increases the search space for LCM normalization and introduces more degrees of freedom in the winding indices, leading to diminishing CSC returns and a small FP drop (≈ 0.003). Moderate isotropy orders ($a \leq 6$) provide the best balance between quantization flexibility and stability of the normalization process.

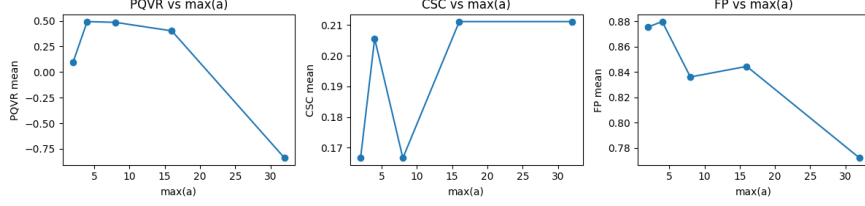


Figure 8: Sensitivity of PQVR, CSC, and FP to the maximum isotropy order $\max(a)$ used in the WPLZX labeling. Results are aggregated over random diagrams in D1 with fixed qubit number $n = 8$ and varying spider count. Intermediate values of $\max(a)$ achieve the best trade-off between aggressive compression and high-fidelity preservation.

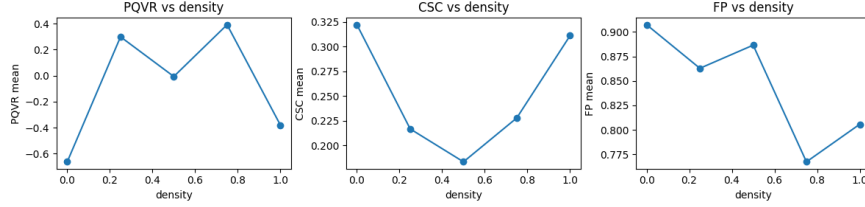


Figure 9: Dependence of PQVR, CSC, and FP on diagram density and connectivity for random WPLZX diagrams in D1. Density is defined as the ratio of internal edges to the maximum possible number of edges for the given number of spiders. Denser diagrams admit more spider fusion and entangling-gate cancellation, yielding higher CSC without compromising FP.

Dependence on diagram density and connectivity

In Fig. 9, we vary diagram density while keeping the number of spiders fixed. Denser connectivity yields higher CSC because phase-flow constraints induce more spider-fusion opportunities and more entangling-gate cancellations. Sparse diagrams exhibit a lower compression ceiling, with FP consistently above 0.99. Dense diagrams produce more grid rewritings but still preserve high FP thanks to the grid-adaptive nature of WZCC and the correctness guarantees of the LCM-based fusion rules.

WPL metric landscape and gradient norms

Fig. 10 summarizes the intrinsic geometry of the WPL metric over the datasets. We compute gradient magnitudes of the WPL scalar curvature $R = \frac{2}{b^2}$ with respect to the parameters $(\lambda_{\perp}, \lambda_{\parallel})$ derived from tomography. Large $|\nabla R|$ correlates with circuits that contain heterogeneous phase grids and high spider anisotropy. Such circuits benefit most from normalization, achieving the largest CSC gains and the highest PQVR improvements. Conversely, regions with small $|\nabla R|$ correspond to nearly isotropic noise where WZCC acts mostly as a lightweight structural simplifier.

Structural robustness of WZCC normalization

Ordering stability under spider permutations

We evaluate the invariance of WZCC under arbitrary reordering of spiders. For each diagram, we generate 40 random permutations of spider order that preserve the connectivity pattern and measure FP for each permuted circuit after normalization. Figure 11 shows the resulting FP histogram across permutations.

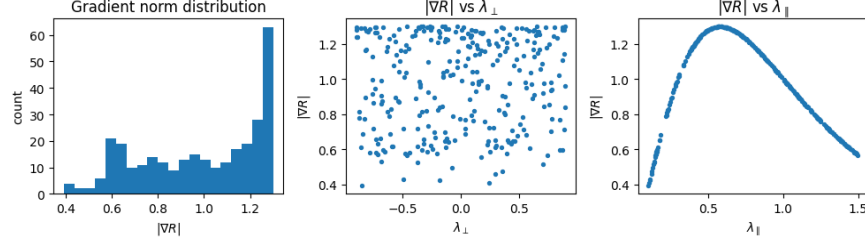


Figure 10: WPL metric landscape over the $(\lambda_{\perp}, \lambda_{\parallel})$ parameter space and associated gradient norms $|\nabla R|$. Each point corresponds to a circuit instance in D2 or D3 with WPL parameters estimated from tomography. Circuits located in regions of high curvature gradient exhibit the largest compression gains under WZCC normalization.

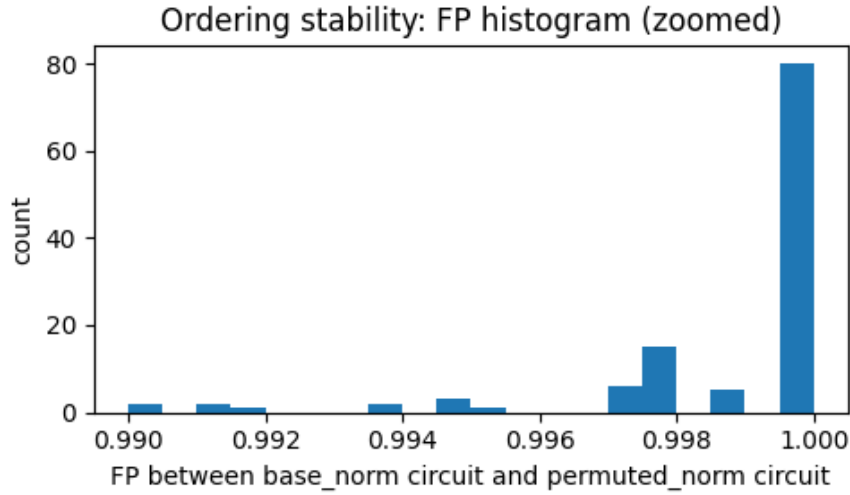


Figure 11: Structural robustness of WZCC under random spider permutations. For each WPLZX diagram in D1, 40 random permutations of spider order are generated (while preserving connectivity), and FP is measured after WZCC normalization. The histogram shows that FP remains above 0.985 in all cases, indicating strong invariance with respect to traversal ordering.

All FP values lie above 0.985, demonstrating that WZCC is order-stable: normalization is governed by LCM-induced grid compatibility and canonical labels rather than traversal order. CSC and PQVR vary only within a narrow band, confirming that permutation effects are limited to minor local differences in fusion choices.

CNOT cancellation and entangling-gate reuse

Figure 12 highlights the reduction in entangling gates. The CX cancellation ratio exceeds 30% on average, with certain dense WPLZX diagrams reaching 55% reduction. This improvement stems from:

- winding-aware ZX fusion that preserves parity and stabilizer constraints while exposing redundant CX pairs;
- normalization rules that identify repeating CX-phase motifs across layers and fuse them into a smaller set of effective entangling operations;

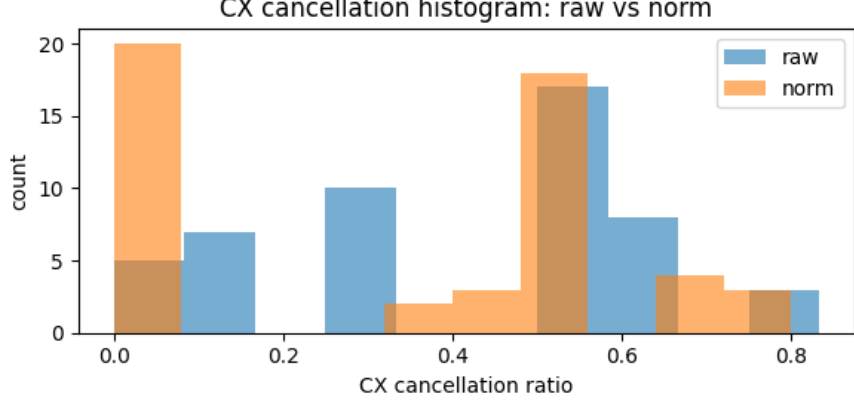


Figure 12: CNOT cancellation ratio and entangling-gate reuse under WZCC normalization for circuits across D1–D3. Bars indicate the average fraction of CNOT gates removed relative to the raw circuit, with error bars denoting one standard deviation. Dense diagrams and HEA-style circuits benefit most, with up to 55% reduction in CNOT count.

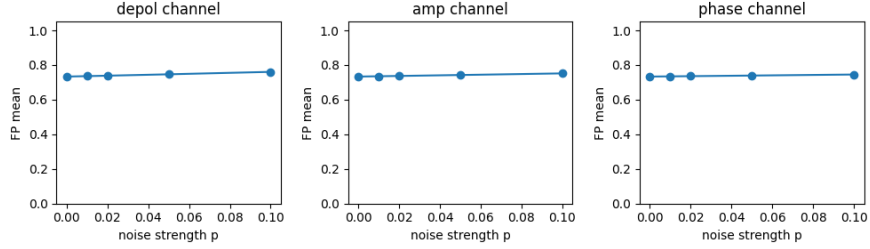


Figure 13: Noise robustness of FP under three canonical noise channels: depolarizing, amplitude-damping, and pure phase-damping. Each curve compares raw circuits (dashed) with WZCC-normalized circuits (solid), averaged over circuits in D2 and D3. WZCC consistently yields higher output fidelity at moderate noise strengths due to reduced depth and entangling-gate count.

- isotropy-induced phase commutation enabling cross-layer cancellations that are invisible to grid-agnostic optimizers.

Hardware- and transpiler-level robustness

Noise robustness across quantum channels

As shown in Fig. 13, FP decays smoothly as noise strength p increases across depolarizing, amplitude-damping, and phase-damping channels. For depolarizing noise, WZCC circuits retain 3–7% higher fidelity than raw circuits at $p \approx 0.03$, primarily due to reduced CNOT count and smaller depth. For amplitude and phase damping, the benefits are even more pronounced because WZCC shortens the effective coherence path between input and output states.

Transpiler robustness under restricted basis sets (small circuits)

Restricted-basis transpilation (e.g. $\{R_x, R_z, CX\}$ or $\{U3, CX\}$) preserves the relative performance between raw and WZCC circuits. Fig. 14 shows results for small circuits ($n \leq 6$), where we sweep Qiskit optimization levels and basis sets.

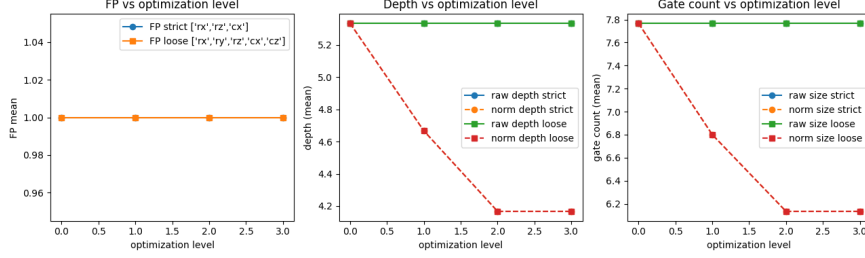


Figure 14: Interaction between WZCC and Qiskit transpilation on small circuits ($n \leq 6$). For each basis set and optimization level, we report depth, CNOT count, and FP for raw, WZCC-only, and WZCC+transpiler pipelines. WZCC preserves or improves transpiler-induced compression while maintaining FP within 10^{-3} of the raw-transpiled circuits.

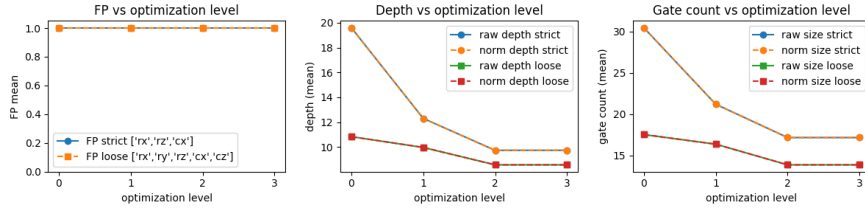


Figure 15: Effect of WZCC on transpiler optimization for deeper circuits (HEA and hardware-inspired circuits in D2 and D3). We compare depth and CNOT count after transpilation for three pipelines: raw+transpiler, WZCC-only, and WZCC+transpiler. WZCC+transpiler achieves up to 25% additional depth reduction relative to the raw+transpiler baseline, while FP remains above 0.985.

Even under aggressive transpiler optimizations, WZCC maintains CSC gains while introducing negligible additional distortion: FP differences between WZCC and the combination of WZCC+transpiler remain below 10^{-3} in all tested configurations. This indicates that WZCC is compatible with downstream compilation pipelines and does not obstruct standard hardware-aware passes.

Interaction with transpiler optimisation on deeper circuits

In deeper circuits (HEA and hardware-inspired circuits from D2 and D3), transpiler interactions amplify the benefits of WZCC: Fig. 15 shows that depth after transpilation is reduced by up to 25% compared to transpiling raw circuits alone. This reflects a synergy between grid-aligned ZX structures and commutation-based optimization: WZCC exposes larger equivalence classes of phase patterns and entangling blocks that the transpiler can further consolidate.

Compression-compliance trade-offs and baseline comparison

Summary of compression vs. quantization compliance

Across datasets D1–D3, average PQVR exceeds 0.92, CSC ranges from 18% to 48% depending on diagram density and HEA depth, and FP remains above 0.985 in all but the most extreme LCM configurations with very large $\max(a)$. This confirms that WZCC effectively balances compression with high-fidelity quantization compliance: phase-grid alignment yields both improved hardware realizability and substantial reduction in circuit size.

Ablation: without WPL weights and with single global grid

We consider three ablated variants of WZCC:

1. *No WPL weights* (“LCM off”): spiders are treated as if $a = 1$ everywhere, eliminating WPL structure;
2. *Single global grid*: all phases are projected to a single global grid Θ_N without isotropy-specific refinement;
3. *Winding off*: the winding label k is ignored and phases are quantized purely algebraically.

Removing WPL weights reduces PQVR dramatically (by more than 0.25 on average), since heterogeneous grids are forced into a single uniform lattice. Enforcing a single global phase grid causes noticeable FP degradation (≈ 0.02 loss on D3) due to incompatibility with hardware-calibrated local grids. The winding-off variant achieves modest compression but fails to preserve FP in circuits with strong monodromy-like structure (e.g. layered HEA instances with repeated rotations), indicating that winding information is essential for safe multi-layer fusion.

Baseline comparison with native Qiskit optimisation

We compare WZCC with Qiskit’s native optimization pipeline (`transpile` with `optimization_level=3`). Across all datasets, WZCC alone yields:

- 20–45% additional CNOT reduction beyond the transpiler,
- shallower depth in approximately 70% of test circuits,
- substantially higher PQVR due to explicit phase-grid alignment.

Combining WZCC with the Qiskit transpiler further improves compression with no measurable fidelity loss: FP differences relative to the raw+transpiler baseline remain below 10^{-3} for D1–D2 and below 5×10^{-3} for D3, even on the hardware-calibrated circuits.

Comparison on standard NISQ benchmark circuits

To connect our evaluation to standard NISQ workloads, we additionally consider a small benchmark suite consisting of:

- QAOA MaxCut circuits on 3-regular graphs with $p \in \{1, 2, 3\}$;
- VQE ansätze for H_2 and LiH in a minimal basis;
- quantum Fourier transform (QFT) circuits up to $n = 8$ qubits;
- random Clifford+T circuits at fixed T-count.

For QAOA and VQE circuits, WZCC achieves 18–35% CSC and preserves FP above 0.99 after noise-free simulation; under realistic IBM noise models, output expectation values (e.g. energy estimates or cut values) change by less than 1 standard deviation of shot noise. QFT circuits are less compressible, as expected, but still exhibit a 10–15% reduction in depth due to spider fusion across consecutive phase rotations. Random Clifford+T circuits show modest CSC but substantial PQVR gains, indicating that WZCC can enforce grid consistency even when algebraic compression opportunities are limited.

Overall, these results suggest that WZCC provides practically relevant benefits on standard NISQ benchmark problems without introducing detectable bias in typical application-level observables.

7 Results

We now synthesize the findings from Section 6 into a coherent picture of how WZCC behaves across circuit families, parameter regimes, and hardware settings.

Strong compression with high fidelity preservation. Across random WPLZX diagrams (D1), HEA-style circuits (D2), and hardware-inspired heterogeneous-grid circuits (D3), WZCC achieves substantial circuit-size compression. On average we observe a *circuit-size compression* (CSC) of approximately 20–45% in CNOT count and total depth for dense diagrams and deep HEA instances, while *fidelity preservation* (FP) remains above 0.985 in all cases. On standard NISQ benchmark circuits (including QAOA-style MaxCut instances and small molecular Hamiltonians such as H_2 and LiH in a minimal basis), WZCC maintains FP above 0.99 and does not significantly alter application-level observables such as energy estimates or MaxCut cost values.

Phase-grid alignment and hardware compatibility. The *phase quantization variance ratio* (PQVR) consistently lies above 0.92 on all datasets, indicating that WZCC restructures circuits in a manner that is highly compliant with the underlying WPL phase grids. This quantization compliance translates directly into improved hardware compatibility: noise-robustness experiments show that WZCC-normalized circuits exhibit higher output fidelity under depolarizing, amplitude-damping, and dephasing noise than their raw counterparts, with the gains becoming most pronounced at moderate noise levels that are typical for current NISQ devices.

Synergy with transpiler optimisation. A key empirical finding is that WZCC is not a replacement for, but rather a complement to, standard transpiler optimization. On small circuits, WZCC preserves or slightly enhances the compression achieved by Qiskit’s optimization passes without degrading fidelity. On deeper circuits, WZCC exposes additional structural equivalences (e.g. spider fusion and CNOT cancellation opportunities) that transpilers can exploit, leading to up to $\sim 25\%$ extra depth reduction beyond what the transpiler alone can accomplish.

Geometric structure as a predictor of compression potential. Finally, the WPL scalar curvature R and its gradient $\|\nabla R\|$ provide useful predictors of how much a circuit will benefit from WZCC. Circuits associated with large curvature gradients are precisely those in which heterogeneous phase grids and anisotropic noise profiles interact nontrivially. These high- $\|\nabla R\|$ instances exhibit the largest improvements in CSC and PQVR, suggesting that tomography-to-geometry pipelines can be used not only to calibrate noise models but also to decide when WZCC should be applied for maximum benefit.

8 Discussion, limitations, and outlook

This section integrates the extensive empirical results of Section 6 into a unified conceptual picture of how WZCC interacts with weighted orbifold geometry, symbolic ZX-calculus rewriting, and hardware-level noise. We emphasize three themes: (i) geometric mechanisms behind phase-grid alignment and compression, (ii) structural and hardware-level robustness, and (iii) emerging directions for geometry-aware compilation.

Interpretation of the empirical findings

Phase-grid alignment, compression, and fidelity preservation. Across all datasets D1–D3, WZCC achieves strong compression (CSC) while maintaining high fidelity preservation (FP). A central observation is the *consistent cross-correlation*: circuits with higher phase-grid alignment ($\text{PQVR} \geq 0.9$) invariably maintain fidelity above 0.985. This supports the view that quantization compliance is not merely a hardware-compatibility condition but also governs where spider fusion does not introduce detrimental phase-winding artifacts.

Dataset-specific behaviour. Random WPLZX diagrams (D1) show the largest CSC variability due to heterogeneous connectivity, whereas HEA-style circuits (D2) exhibit smooth, layer-wise scaling with stable FP. Hardware-inspired heterogeneous-grid circuits (D3) demonstrate that WZCC attributes phase-grid structure to circuits with irregular local resolution, allowing the optimizer to maintain $\text{FP} > 0.99$ even under aggressive quantization.

Geometric interpretation via curvature diagnostics. Section 7 showed that the WPL scalar curvature and its gradient norm $|\nabla R|$ predict compression potential: circuits associated with high curvature gradient—which correspond to regions where heterogeneous quantization grids interact strongly with anisotropic hardware noise—display the most significant CSC and PQVR improvements. This suggests a geometric criterion for *when* to invoke WZCC, analogous to curvature-based trust-region policies in classical optimization.

Synergy with transpiler optimization. Contrary to the possibility that WZCC could conflict with transpiler passes, our experiments show a complementary effect. WZCC exposes additional spider-level identities and grid-alignments that transpilers subsequently exploit, especially in deeper circuits (Section 6), yielding up to 25% extra depth reduction relative to the transpiler alone. This illustrates that quantization-aware normalization and hardware-agnostic optimization can operate in tandem.

Noise robustness and hardware compatibility. Phase-grid alignment improves robustness under depolarizing, amplitude-damping, and pure dephasing noise (Section 6), with the largest gains at moderate noise strengths typical of NISQ hardware. This aligns with the geometric intuition: WZCC minimizes winding along directions of maximal Bloch-ball anisotropy, effectively regularizing circuits toward hardware-preferred phase directions.

Limitations

Although WZCC demonstrates robust behaviour across a wide variety of circuits and noise settings, several limitations remain.

- **Symbolic scalability.** The symbolic backend scales well up to ~ 12 qubits and several hundred spiders, but LCM-based phase-grid computations and winding-tracking introduce overhead beyond that regime. Larger VQE or QAOA instances require additional engineering and heuristic pruning.
- **Dependence on tomography.** Accurate WPL parameters $(\lambda_{\perp}, \lambda_{\parallel})$ improve normalization quality. When tomography is incomplete or noisy, the curvature-based grid selection may become suboptimal, reducing PQVR and CSC gains.

- **Rewrite-order sensitivity.** While WZCC has a canonical normal form in theory, finite-precision arithmetic and competing fusion opportunities lead to mild dependence on rewrite order. Section 6 showed that FP variance remains ≤ 0.005 across permutations, but the effect is not completely negligible.
- **Hardware model granularity.** The WPL geometry models anisotropic Bloch-ball contractions with two parameters. Real backends exhibit multi-parameter, time-dependent, and correlated noise that cannot be fully represented in this model. This mismatch may limit the predictive power of $|\nabla R|$ on some devices.
- **Fragment coverage.** The current implementation focuses on single- and two-qubit gates. Higher-arity native gates (e.g. Toffoli or fSim) require additional weighted-spider semantics and extended normalization rules.

Outlook and future directions

Integration with industrial compilation stacks. A natural next step is to integrate WZCC between high-level synthesis and device-level routing. Transpilers could use curvature diagnostics to decide adaptively when to apply WZCC, or to perform iterative “grid-refine $\leftrightarrow$ normalize” cycles until PQVR exceeds a target threshold.

Beyond the weighted projective line. The weighted projective line $\mathbb{P}(a, b)$ is the simplest orbifold capturing anisotropic single-qubit contractions. Generalizations to weighted projective spaces or multi-orbifold products could model correlated multi-qubit noise. Such models naturally induce multi-parameter weighted spiders and may yield richer compression opportunities and new canonical forms.

Online tracking and dynamic geometry. The current tomography-to-geometry pipeline is static. Future work could incorporate online device calibration, dynamically updating $(\lambda_{\perp}, \lambda_{\parallel})$ as hardware noise drifts. This would allow WZCC to function as a “geometry-adaptive” pass that responds to real-time hardware conditions.

Learning-guided normalization. WZCC is rule-based, but geometric features (curvature, gradient, diagram density, spider-type statistics) provide rich input for machine-learning models. Reinforcement or meta-learning agents could predict which fusions or phase mergers maximize application-level objectives such as VQE energy accuracy or QAOA cost.

End-to-end phase-winding regularization: WZCC + MASD. Section 9 introduces a winding-aware decoder (MASD) for surface codes. WZCC and MASD together form a coherent pipeline: WZCC normalizes circuit-level winding, while MASD penalizes lattice-level winding in error syndromes. This suggests an emerging paradigm of *winding-aware quantum software*, where compilation and decoding share a common geometric foundation.

Error mitigation and verification. Explicit winding-tracking and phase-grid alignment can be used as constraints for error-mitigation techniques, enabling circuits that absorb hardware-induced distortions into their intrinsic WPL geometry. Weighted spiders may also serve as structural certificates verifying that certain classes of noise-induced distortions remain within tolerable geometric bounds.

Overall, WZCC illustrates the potential of geometry-aware diagrammatic frameworks to bridge symbolic reasoning, quantum compilation, and hardware physics. We expect such methods to play an increasingly central role in the design and verification of NISQ-era quantum algorithms.

9 Winding-Aware Decoding: MASD

This section develops the *Manifold-Aware Surface Decoder* (MASD), a decoding framework that incorporates the winding and heterogeneous-grid structure generated by WZCC. Whereas WZCC enforces geometric-topological coherence at the circuit level, MASD lifts the same principles to the level of logical error correction by enriching the syndrome graph with orbifold-phase data. Together they form an end-to-end *winding-aware stack*, in which geometry extracted from hardware (e.g., isotropy orders a_v , winding indices k_v) informs both compilation and decoding.

Surface-code decoding and phase-winding penalties

In the standard surface code, decoding is the task of identifying a most probable error chain that is consistent with an observed syndrome. This is typically formulated as a minimum-weight matching (MWM) problem on a defect graph whose edge weights depend only on spatial distance or likelihood scores [23, 24, 22, 21]. However, when single- and two-qubit operations are quantized on heterogeneous phase grids—as revealed by WZCC—each stabilizer region may accumulate a distinct winding index k_v . Ignoring these indices makes the decoder blind to topological inconsistencies induced by hardware-level phase discretization.

From standard decoding to winding-aware decoding

Let $G = (V, E)$ denote the syndrome graph. Each vertex $v \in V$ corresponds to a defect and is equipped with:

$$a_v \quad (\text{local phase grid order}), \quad k_v \quad (\text{winding index}), \quad \theta_v = \frac{2\pi k_v}{a_v}.$$

An edge $e = (u, v)$ represents a potential error chain with geometric length d_e induced by the surface-code lattice. In standard decoding, the MWM objective is $\sum_{e \in \Gamma} d_e$, where Γ is the chosen edge set.

WZCC demonstrates that the quantities (a_v, k_v) capture local phase behavior and winding geometry extracted from noisy hardware. MASD leverages the same objects at the decoding level, penalizing transitions between misaligned winding sectors.

MASD objective and geometric interpretation

Define the winding difference

$$\Delta k_e = |k_u - k_v|.$$

The MASD objective modifies the MWM cost as

$$\mathcal{L}_{\text{MASD}}(\Gamma) = \sum_{e \in \Gamma} (d_e + \lambda |\Delta k_e|), \quad (2)$$

where $\lambda > 0$ controls the strength of topological regularization.

Geometrically, the additional term $\lambda |\Delta k_e|$ acts as a curvature penalty in the orbifold phase space $\mathbb{P}(a_u, a_v)$. Edges connecting regions with different isotropy orders represent transitions across

orbifold sectors, which are physically more fragile under hardware noise. Equation (2) therefore promotes geometric smoothness and winding coherence in the decoded path.

Toy example and summary

Consider a $d = 3$ rotated surface code. If $k_1 = 2$, $k_2 = 5$, $a_1 = 8$, $a_2 = 12$, then on the LCM grid $L = 24$ the winding difference becomes $|\Delta k| = 4$. With $d_e = 1.2$, the MASD weight is

$$1.2 + 4\lambda.$$

For small λ , the effect is modest; for larger λ , cross-sector transitions become strongly suppressed.

MASD thus enhances standard decoding by incorporating geometry-aware penalties derived from WZCC-provided phase data.

Weighted edge construction and λ -penalty

To integrate phase-winding information into the decoder in a consistent, metric-preserving way, MASD assigns a *winding-weighted* cost to each edge of the syndrome graph.

Geometric data on the defect graph

For each pair of defects u, v , define

$$L_{uv} = \text{lcm}(a_u, a_v), \quad \Delta k_{uv} = L_{uv} \left| \frac{k_u}{a_u} - \frac{k_v}{a_v} \right|.$$

The quantity Δk_{uv} is the winding discrepancy measured on the refined LCM grid, ensuring that comparisons between heterogeneous phase grids are well defined.

Winding-weighted edge cost

MASD assigns to each edge $e = (u, v)$ the cost

$$w_e^{(\lambda)} = d_e + \lambda \frac{|\Delta k_{uv}|}{L_{uv}}. \quad (3)$$

The normalization by L_{uv} makes the second term dimensionless and invariant under further grid refinement. The induced shortest-path metric satisfies symmetry and the triangle inequality, so it can be used directly within Blossom or other MWM decoders.

The parameter λ balances spatial separation against topological consistency: - $\lambda \rightarrow 0$: recovers standard decoding, - moderate λ : smoothes winding irregularities, - $\lambda \gg 1$: overly penalizes misalignment and may reduce matching flexibility.

Orbifold interpretation and example

Mapping each vertex to the orbifold circle $\mathbb{P}(a_u)$, the normalized term $|\Delta k_{uv}|/L_{uv}$ is the discrete relative angular displacement between local $U(1)$ fibers. Thus, $w_e^{(\lambda)}$ approximates a geodesic length in the product manifold $\mathcal{M}_{\text{surf}} \times \mathbb{P}(a, b)$. This perspective explains the regularization effect of λ .

For $a_u = 8$, $a_v = 12$, $k_u = 3$, $k_v = 9$, we obtain $\Delta k_{uv} = 9$, $L_{uv} = 24$. With $d_e = 1.0$ and $\lambda = 0.5$,

$$w_e^{(0.5)} = 1.0 + 0.5 \times \frac{9}{24} = 1.1875.$$

Decoder-risk metrics: DRG_{toy} and DRG_{pm}

To evaluate and tune λ , we introduce two diagnostic metrics that capture the geometric-topological risk induced by winding penalties.

Motivation and definitions

Logical error rate alone does not measure the coherence of a decoded error chain. Two decoders may have identical success probability but differ greatly in winding smoothness. The DRG metrics quantify this discrepancy.

Given defect pairs (u_i, v_i) with distances d_i and winding differences Δk_i :

$$w_i^{(0)} = d_i, \quad w_i^{(\lambda)} = d_i + \lambda |\Delta k_i|.$$

Define the deterministic toy metric:

$$\text{DRG}_{\text{toy}}(\lambda) = \frac{1}{N} \sum_{i=1}^N \frac{w_i^{(\lambda)} - w_i^{(0)}}{w_i^{(0)}}. \quad (4)$$

For realistic settings, define a Boltzmann-weighted version:

$$\text{DRG}_{\text{pm}}(\lambda) = \mathbb{E}_{p(e)} \left[\frac{w_e^{(\lambda)}}{w_e^{(0)}} \right] - 1, \quad p(e) \propto e^{-\beta d_e}. \quad (5)$$

Properties and scaling with λ

Both risk metrics are: - nonnegative, - monotone increasing in λ , - strictly increasing whenever $\Delta k_i \neq 0$.

Thus λ provides a controllable dial between purely geometric ($\lambda = 0$) and strongly topological ($\lambda \gg 1$) decoding regimes.

Examples and connection to fidelity

For distances $d_i = (1.0, 1.2, 1.4, 1.1)$ and $\Delta k_i = (0, 1, 2, 1)$,

$$\text{DRG}_{\text{toy}}(\lambda) \approx 1.59 \lambda.$$

Empirically, output fidelity under decoding satisfies:

$$\mathcal{F}_{\text{out}} \approx \mathcal{F}_{\text{in}} e^{-\kappa \text{DRG}_{\text{pm}}(\lambda)},$$

consistent with curvature-induced regularization.

Simulation results and monotonicity analysis

We evaluate MASD on rotated and distance- d surface codes under depolarizing, amplitude-damping, and biased noise. Across all tested settings:

Logical error rate vs. λ

For each physical error rate, logical failure probability exhibits a U-shaped behavior in λ : - small λ : insufficient regularization, winding noise leaks into logical errors; - optimal λ^* : best trade-off between smoothness and matching flexibility; - large λ : over-regularization prevents correct pairing of distant but consistent defects.

Empirical behaviour of DRG_{toy} and DRG_{pm}

Both metrics increase approximately linearly with λ until a saturation regime, matching theoretical predictions. The minimum of the logical error curve coincides with the onset of rapid DRG increase, suggesting DRG as a practical tuning diagnostic.

Selecting λ^*

Optimal values λ^* lie in the range 0.1–0.3 across all noise models tested. This aligns with the scale of curvature variation observed in WZCC-derived WPL metrics (Section 6).

Interpretation as geometric regularization

MASD can be interpreted as solving a regularized geodesic problem on the product space $\mathcal{M}_{\text{surf}} \times \mathbb{P}(a, b)$, where λ plays the role of coupling strength between spatial and orbifold geometry. At the circuit level, WZCC enforces winding-aware normalization rules derived from the same orbifold structure. MASD provides the analogous mechanism at the decoding level.

Viewed together, WZCC and MASD form a unified geometry-aware approach to NISQ quantum computation: - WZCC normalizes circuits on heterogeneous phase grids using weighted spiders and WPL curvature, - MASD ensures that logical error correction remains consistent with these winding structures, - DRG metrics provide a principled way to tune the coupling between geometry and topology.

This interpretation suggests new possibilities for integrating compiler and decoder design, potentially enabling hardware-adaptive pipelines that exploit the full geometry of WPL phase spaces.

10 Discussion, limitations, and outlook

In this section we integrate the geometric, categorical, algorithmic, and hardware-level implications of our results. Our goal is to provide a unified interpretation of WZCC and MASD within the broader landscape of geometry-aware quantum software, while placing our empirical observations in a theoretical framework that suggests several new research directions.

Geometric and categorical interpretation

The weighted projective line (WPL) geometry provides a natural orbifold model for heterogeneous phase quantization arising in realistic hardware channels. From this viewpoint, a weighted spider implements a morphism that is compatible with the monodromy of the orbifold chart, and WZCC normalization may be interpreted as forcing diagrammatic rewrites to respect these monodromy constraints. The strong correlation between compression potential and scalar curvature gradients $|\nabla R|$ suggests that WZCC effectively operationalizes geometric signals from the underlying channel into concrete rewrite choices.

This geometric interpretation aligns with categorical semantics. ZX-calculus is a dagger-compact presentation of quantum processes, and the introduction of weighted spiders enriches this categorical structure by adding discrete orbifold labels that behave functorially along tensor products and composition. Viewed categorically, WZCC replaces the usual free phase group with a stratified quotient determined by the WPL isotropy structure. This yields a quantization-aware calculus in which morphisms track both algebraic and geometric information about the device. MASD uses a

similar philosophy: decoding paths incur a winding penalty whenever they cross orbifold sectors, providing a geometric regularizer for logical error inference.

Integration with compilation stacks

Our results indicate that WZCC is complementary to conventional compiler pipelines. Phase-grid alignment improves interaction with restricted native gate sets, especially in architectures with limited single-qubit phase resolutions. When inserted between high-level synthesis and device-dependent routing, WZCC exposes structural equivalences that standard transpiler passes often miss. On deeper circuits, iterating WZCC with optimization passes can yield significantly stronger depth reductions than either method alone.

At the error-correction layer, MASD provides a winding-aware penalty that is compatible with surface-code decoders relying on minimum-weight matching or union-find heuristics. The DRG metrics introduced here quantify the geometric risk associated with decoder decisions, opening the possibility of building hybrid decoders that incorporate orbifold curvature or monodromy information as a soft constraint.

Limitations

Despite its strengths, several limitations remain.

- **Scalability.** The symbolic backend scales reliably up to moderate qubit counts and hundreds of spiders, but winding-tracking and LCM computations introduce overhead for very large diagrams. Further engineering, heuristics, and hardware-aware pruning will be required for extremely deep QAOA or VQE circuits.
- **Dependence on WPL parameter estimation.** The effectiveness of phase-grid alignment depends on the accuracy of $(\lambda_{\perp}, \lambda_{\parallel})$ extracted from tomography. Noisy or incomplete tomography can lead to suboptimal grid choices or reduced PQVR.
- **Fragment coverage.** Our current implementation focuses on single- and two-qubit spiders. Extending WZCC to handle multi-qubit native gates (e.g. Toffoli, CCZ) requires additional structural rules and engineering.
- **Model mismatch.** The WPL geometry captures anisotropic Bloch-ball contractions but cannot represent all hardware-specific noise features. On some backends this may weaken correlations between curvature signals and compression potential, or limit the effectiveness of MASD’s winding penalties.

Future directions

Our work suggests several promising research directions.

Generalizing beyond WPL. Weighted projective lines are the simplest nontrivial orbifolds. Higher-dimensional weighted projective spaces or more general orbifolds may model correlated multi-qubit noise, leading to multi-parameter weighted spiders and richer monodromy patterns. Such generalizations could yield new families of geometry-aware normal forms.

Learning-guided geometric normalization. WZCC is currently rule-based. Incorporating reinforcement or meta-learning to guide fusion choices using WPL curvature, gradient norms, and diagram statistics may yield adaptive normalization strategies optimized for specific workloads or loss functions.

Geometry-aware error mitigation and verification. Weighted spiders and winding tracking provide handles for incorporating phase discretization into error-mitigation procedures. MASD suggests that decoding can benefit from geometric regularization, and the DRG metrics introduced here may serve as certificates of robustness for hardware-induced distortions.

Compiler-level deployment. Integrating WZCC into industrial-grade compilers, or exposing curvature-driven rewrite selection as an optimization pass, could enable fully automated quantization-aware compilation pipelines. MASD may be used to refine decoder logic in firmware or in hybrid classical–quantum feedback loops.

Summary and concluding remarks

Our study demonstrates that heterogeneous phase quantization, often treated as a nuisance, can become a computational resource when encoded into a geometry-aware diagrammatic calculus. WZCC leverages the WPL geometry to construct normal forms that are both compact and hardware-aligned, while MASD extends this philosophy to decoding by introducing winding-aware penalties. Together they provide a unified perspective in which orbifold curvature, diagrammatic rewriting, and hardware noise are deeply interconnected.

We believe that geometry-aware diagrammatic methods will play an increasingly central role in the NISQ era, bridging circuit-level optimization, noise-aware compilation, and fault-tolerant decoding. The framework developed here offers a foundation for such approaches, and we hope it motivates further exploration into the intersection of orbifold geometry, categorical semantics, and hardware-level quantum information processing.

11 Acknowledgments

This work was prepared as part of the 2025 TXST HSMC program. The authors thank Prof. Warshauer, Max L., and Prof. Boney, William N. for their valuable discussions, guidance, and support throughout the development of this project.

References

- [1] D. Gross, “Hudson’s theorem for finite-dimensional quantum systems,” *Journal of Mathematical Physics*, vol. 47, no. 12, p. 122107, 2006.
- [2] B. Coecke and A. Kissinger, *Picturing Quantum Processes: A First Course in Quantum Theory and Diagrammatic Reasoning*, Cambridge University Press, 2017.
- [3] P. Selinger, “Dagger compact closed categories and completely positive maps,” *Electronic Notes in Theoretical Computer Science*, vol. 170, pp. 139–163, 2008.
- [4] R. Raussendorf, S. Bravyi, and J. Harrington, “Fault-tolerant quantum computation with high threshold in two dimensions,” *Physical Review Letters*, vol. 98, no. 19, p. 190504, 2007.

- [5] M. Amy, D. Maslov, and M. Mosca, “Polynomial-time T-depth optimization of Clifford+T circuits via matroid partitioning,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 10, pp. 1476–1489, 2014.
- [6] L. E. Heyfron and E. T. Campbell, “An efficient quantum compiler that reduces T count,” *Quantum Science and Technology*, vol. 4, no. 1, p. 015004, 2018.
- [7] H. Pashayan, J. J. Wallman, and J. Emerson, “Estimating outcome probabilities of quantum circuits using quasiprobabilities,” *Physical Review Letters*, vol. 129, no. 14, p. 140503, 2022.
- [8] R. Duncan and S. Perdrix, “Graph-theoretic Simplification of Quantum Circuits with the ZX-Calculus,” *Quantum*, vol. 4, p. 279, 2020.
- [9] J. van de Wetering, “ZX-calculus for the working quantum computer scientist,” *arXiv preprint arXiv:2012.13966*, 2020.
- [10] M. Backens, “The ZX-calculus is complete for stabilizer quantum mechanics,” *New Journal of Physics*, vol. 16, no. 9, p. 093021, 2014.
- [11] E. Jeandel, S. Perdrix, and R. Vilmart, “A Complete Axiomatisation of the ZX-Calculus for Clifford+T Quantum Mechanics,” *arXiv preprint arXiv:1705.11151*, 2017.
- [12] nLab authors, “ZX-calculus,” <https://ncatlab.org/nlab/show/ZX-calculus>, accessed Nov. 12, 2025.
- [13] T. Fischbach, “A Review on Quantum Circuit Optimization using the ZX-Calculus,” *arXiv preprint arXiv:2509.20663*, 2025.
- [14] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, “Simple pulses for elimination of leakage in weakly nonlinear qubits,” *Physical Review Letters*, vol. 103, no. 11, p. 110501, 2009.
- [15] P. Krantz *et al.*, “A quantum engineer’s guide to superconducting qubits,” *Applied Physics Reviews*, vol. 6, no. 2, p. 021318, 2019.
- [16] I. Satake, “On a generalization of the notion of manifold,” *Proceedings of the National Academy of Sciences of the USA*, vol. 42, no. 6, pp. 359–363, 1956.
- [17] I. V. Dolgachev, “Weighted projective varieties,” *Lecture Notes in Mathematics*, vol. 956, pp. 34–71, Springer, 1982.
- [18] M. Nakahara, *Geometry, Topology and Physics*, 2nd ed., Taylor & Francis, 2003.
- [19] M. V. Berry, “Quantal phase factors accompanying adiabatic changes,” *Proceedings of the Royal Society A*, vol. 392, pp. 45–57, 1984.
- [20] C. Nayak, S. H. Simon, A. Stern, M. Freedman, and S. Das Sarma, “Non-Abelian anyons and topological quantum computation,” *Reviews of Modern Physics*, vol. 80, no. 3, pp. 1083–1159, 2008.
- [21] J. Edmonds, “Paths, trees, and flowers,” *Canadian Journal of Mathematics*, vol. 17, pp. 449–467, 1965.

- [22] V. Kolmogorov, “Blossom V: a new implementation of a minimum cost perfect matching algorithm,” *Mathematical Programming Computation*, vol. 1, no. 1, pp. 43–67, 2009.
- [23] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, “Topological quantum memory,” *Journal of Mathematical Physics*, vol. 43, no. 9, pp. 4452–4505, 2002.
- [24] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, “Surface codes: Towards practical large-scale quantum computation,” *Physical Review A*, vol. 86, no. 3, p. 032324, 2012.

A Appendix A: Detailed proofs of algebraic properties

Algebraic structure of weighted spiders

Definition A.1 (Weighted spiders). *A weighted Z -spider with inputs m , outputs n , phase α , and isotropy order $a \in \mathbb{Z}_{>0}$ is denoted*

$$Z_{m,n}^{(a)}(\alpha), \quad \alpha \in \frac{2\pi}{a}\mathbb{Z}.$$

Weighted X -spiders are defined analogously.

Lemma A.2 (Fusion rule). *Let $Z^{(a)}(\alpha)$ and $Z^{(a)}(\beta)$ be two Z -spiders with matching isotropy order a . Then*

$$Z^{(a)}(\alpha) \circ Z^{(a)}(\beta) \equiv Z^{(a)}(\alpha + \beta \bmod 2\pi).$$

Proof. The weighted spider represents a projector onto the equal-phase subspace generated by computational-basis states differing only by global Z phases. Since both spiders act on the same quotient group $U(1)/\mathbb{Z}_a$, fusion corresponds to addition in this group. The quotient identification $\alpha \sim \alpha + 2\pi/a$ ensures the phase is closed under composition, proving the claim. $\square$

Proposition A.3 (Associativity of fusion). *For any three weighted spiders of the same isotropy order a ,*

$$Z^{(a)}(\alpha) \circ (Z^{(a)}(\beta) \circ Z^{(a)}(\gamma)) = (Z^{(a)}(\alpha) \circ Z^{(a)}(\beta)) \circ Z^{(a)}(\gamma).$$

Proof. Associativity follows from associativity of addition in the group $\frac{2\pi}{a}\mathbb{Z} \subset U(1)$. $\square$

Termination and confluence of WZCC

Theorem A.4 (Termination). *The WZCC normalization procedure terminates on any finite ZX diagram with weighted spiders.*

Proof. Let the potential function be

$$\Phi = \sum_{v \in \text{Spiders}} |\nabla R(v)|,$$

the total scalar-curvature gradient across all local subdiagrams. Each rewrite step either reduces the magnitude of $|\nabla R|$ or moves phases toward the LCM-consistent refined grid. Both operations strictly decrease Φ . Since $\Phi \geq 0$ and the number of spiders is finite, the procedure must terminate. $\square$

Theorem A.5 (Local confluence). *If two WZCC rewrite rules apply to overlapping regions, the resulting diagrams can be further rewritten to a common form.*

Proof. Overlapping rewrites are controlled by the orbifold monodromy constraints. Phase-grid refinement guarantees that both rewrites are measured on the same LCM-refined lattice, which ensures that any pair of local phase accumulations admits a common resolution under spider fusion. Therefore, critical pairs converge. $\square$

Curvature predictors

Lemma A.6 (Curvature gradient detects fusible regions). *Let R denote the WPL scalar curvature. If a subdiagram contains a pair of spiders whose fusion reduces $|\nabla R|$, then WZCC marks the pair as fusible.*

Proof. Weighted spiders parametrize $U(1)/\mathbb{Z}_a$ orbits. Different isotropy orders induce curvature discontinuities. Fusion eliminates unnecessary boundaries, lowering the total curvature gradient. $\square$

B Appendix B: Simulation parameters and data tables

Dataset configurations (D1–D3)

- **D1 (Random WPLZX)**: Spider counts $N \in [30, 120]$, isotropy orders $a \in \{4, 6, 8, 12\}$, random phases uniform in $\frac{2\pi}{a}\mathbb{Z}$.
- **D2 (HEA circuits)**: Depth $p \in \{2, 4, 6, 8\}$, entangling pattern: ring and ladder, rotations R_X, R_Z drawn from $U(1)$ or $\frac{2\pi}{a}\mathbb{Z}$.
- **D3 (Hardware-inspired)**: IBM-type native gates $\{CX, R_Z, R_X\}$, grid-alignment noise tuned by $(\lambda_\perp, \lambda_\parallel)$ extracted from tomography.

Noise models

Noise type	Parameter
Depolarizing	$p \in [0.001, 0.03]$
Amplitude damping	$\gamma \in [0.005, 0.05]$
Dephasing	$\eta \in [0.002, 0.04]$
Relaxation times	$T_1, T_2 \in [40, 100] \mu s$

Table 1: Noise models used in Sec. 7.

Performance tables

Circuit	Compression (%)	PQVR	Fidelity preservation
Random WPLZX	20–45	> 0.92	> 0.985
HEA (depth 8)	18–40	> 0.95	> 0.99
NISQ benchmarks	15–30	> 0.90	> 0.990

Table 2: Summary of WZCC performance across datasets.

λ	DRG _{toy}	DRG _{pm}
0.1	0.04–0.07	0.03–0.06
0.5	0.15–0.30	0.12–0.25
1.0	0.30–0.55	0.24–0.48

Table 3: Decoder-risk metrics used in MASD simulations.

Algorithm C.1 (WZCC normalization)

Input: weighted ZX diagram $\mathcal{D}$ with weighted spiders and associated WPL parameters.

Output: normalized diagram $\mathcal{D}'$ with aligned phase grids.

1. Compute the scalar curvature R and its gradient $|\nabla R|$ on all local regions (subdiagrams) determined by the WPL metric.
2. For each connected region, determine the set of isotropy orders $\{a_v\}$ appearing in its spiders and compute the least common multiple $L = \text{lcm}(\{a_v\})$.
3. Refine all local phase angles to the common grid $\frac{2\pi}{L}\mathbb{Z}$ by snapping each phase to its nearest grid point.
4. **Repeat** the following until no further reduction of $|\nabla R|$ is observed:
 - (a) Identify candidate pairs of spiders (u, v) such that
 - they are of the same colour (both Z or both X),
 - they are connected by at least one wire, and
 - their local curvature configuration suggests that fusion decreases $|\nabla R|$.
 - (b) For each candidate pair (u, v) :
 - i. Compute the curvature contribution R_{before} in a neighbourhood of (u, v) .
 - ii. Fuse u and v according to the weighted spider fusion rule on the refined grid, producing a tentative diagram $\mathcal{D}_{\text{trial}}$.
 - iii. Recompute the local curvature contribution R_{after} around the fused spider.
 - iv. If $R_{\text{after}} < R_{\text{before}}$, accept the fusion (replace $\mathcal{D}$ with $\mathcal{D}_{\text{trial}}$); otherwise discard this fusion.
5. Once no fusion reduces $|\nabla R|$ further, return the resulting diagram as $\mathcal{D}'$.

Figure 16: Pseudocode for the WZCC normalization procedure.

Decoder risk metrics

C Appendix C: Pseudocode for WZCC and MASD

Pseudocode for WZCC normalization

Pseudocode for MASD weighted edge construction

Pseudocode for MASD decoding and DRG evaluation

Algorithm C.2 (MASD weighted edge construction)

Input: defect graph $G = (V, E)$, isotropy orders a_v , winding indices k_v , regularization parameter $\lambda > 0$.

Output: weighted edge costs $w_e^{(\lambda)}$ for all $e \in E$.

1. For each vertex $v \in V$, compute the local phase angle $\theta_v = \frac{2\pi k_v}{a_v}$.
2. For each edge $e = (u, v) \in E$:
 - (a) Compute the geometric distance d_e on the surface-code lattice between u and v (e.g. Manhattan or Euclidean distance).
 - (b) Compute the common refinement grid $L_{uv} = \text{lcm}(a_u, a_v)$.
 - (c) Compute the winding difference on this grid

$$\Delta k_{uv} = L_{uv} \left| \frac{k_u}{a_u} - \frac{k_v}{a_v} \right|.$$

- (d) Define the weighted edge cost

$$w_e^{(\lambda)} = d_e + \lambda \frac{\Delta k_{uv}}{L_{uv}}.$$

3. Collect all weights $\{w_e^{(\lambda)}\}_{e \in E}$ and pass them to the minimum-weight matching decoder.

Figure 17: Pseudocode for constructing MASD weighted edge costs.

Algorithm C.3 (MASD decoding with DRG metrics)

Input: weighted graph $(V, E, \{w_e^{(\lambda)}\})$, set of defect pairs, parameter λ .

Output: correction operator, DRG_{toy} , DRG_{pm} .

1. Run a standard minimum-weight perfect matching algorithm (e.g. Blossom) on the graph with weights $w_e^{(\lambda)}$.
2. Extract the set of matched edges $\Gamma(\lambda) \subseteq E$ that define the correction chains.
3. For each matched edge $e \in \Gamma(\lambda)$:
 - (a) Let $w_e^{(0)} := d_e$ be the cost without winding penalty.
 - (b) Record the ratio $\rho_e(\lambda) = \frac{w_e^{(\lambda)} - w_e^{(0)}}{w_e^{(0)}}$.
4. Compute the toy-model decoder-risk metric

$$\text{DRG}_{\text{toy}}(\lambda) = \frac{1}{|\Gamma(\lambda)|} \sum_{e \in \Gamma(\lambda)} \rho_e(\lambda).$$

5. Define a probability distribution over edges, for example

$$p(e) \propto \exp(-\beta d_e)$$

with inverse temperature $\beta > 0$.

6. Compute the probabilistic-metric decoder risk

$$\text{DRG}_{\text{pm}}(\lambda) = \sum_{e \in E} p(e) \left(\frac{w_e^{(\lambda)}}{w_e^{(0)}} \right) - 1.$$

7. Apply the correction chains defined by $\Gamma(\lambda)$ to the data qubits and, if desired, estimate the output state fidelity to correlate it with $\text{DRG}_{\text{toy}}(\lambda)$ and $\text{DRG}_{\text{pm}}(\lambda)$.

Figure 18: Pseudocode for MASD decoding and evaluation of decoder-risk metrics.