Synergy-Guided Refraction Map (Qrack)

A Walkthrough, Use Cases, and Future Directions

Supplement to `synergy_refraction_qrack_stable.py`

1 Purpose and High-Level Flow

This benchmark isolates the effect of a *deterministic refraction mask* and an optional *coherence-weighted truncation* (CWT) pass in Qrack. It runs four configurations on the *same* seeded random circuit:

1. **REF**: tighter rounding, no refraction, CWT off (reference distribution).
2. **BASE**: aggressive rounding, no refraction, CWT off (baseline speed).
3. **BASE+R**: aggressive rounding, *REF-derived* refraction mask, CWT off.
4. **MOD**: aggressive rounding, *REF-derived* refraction mask, CWT on.

The refraction mask is computed *once* from the REF samples and reused for BASE+R and MOD, so MOD–vs–BASE+R isolates the CWT effect. A post-REF *calculation layer* turns REF neighbor labels into a compact, rotation-invariant signature:

- phase-lock shift k (best cyclic alignment),
- correlation scores (C, C_w),
- balanced-ternary difference vector and integer D.

2 Environment and Parameters (unchanged knobs)

Core run knobs

$$N = 20, \quad \mathrm{DEPTH} = 24, \quad \mathrm{SHOTS} = 10{,}000, \quad \mathrm{SEED} = 12345$$

Ring entanglers; sparse Hadamards; small-angle R_x/R_y sprinkles.

Label thresholds Neighbor Pearson correlations r_i are mapped to labels $\ell_i \in \{+1, 0, -1\}$ with

$$\tau_+ = +0.005, \quad \tau_- = -0.005, \quad \ell_i = \begin{cases} +1 & r_i \geq \tau_+ \\ -1 & r_i \leq \tau_- \\ 0 & \text{otherwise.} \end{cases}$$

Refraction policy

KEEP_SCALE_ALPHA = 1.00, KEEP_SCALE_ZERO = 0.00, KEEP_SCALE_BETA = 0.00, SKIP_CX_BET

Small-rotation threshold $T_{\mathrm{TN}} = 10.9$ (gates with $|\theta| < T_{\mathrm{TN}}$ are candidates for keep/skip according to per-qubit keep probabilities).

Rounding & CWT

REF rounding $= 0.020$, aggressive rounding = `QRACK_NONCLIFFORD_ROUNDING_THRESHOLD` $= 0.60$.

CWT toggled via `QRACK_COHERENCE_TRUNC` $\in \{0, 1\}$ with $\lambda =$ "10.12", maxboost = "100.08".

Key environment pins

- `QRACK_OCL_DEFAULT_DEVICE=0` (pin to NVIDIA GPU 0).
- `QRACK_QTENSORNETWORK_THRESHOLD_QB=64` to avoid TN in REF/pilot.
- `QRACK_DISABLE_QUNIT_FIDELITY_GUARD=1`, separability threshold 0.18.

3 Circuit Construction

The circuit builder creates a reproducible gate list (seeded):

- **H layer**: sparse, here H_prob $= 1.0$ (full layer).
- **Rotation sprinkles**: per qubit per layer, with rates RX_RATE = RY_RATE = 0.85 and magnitude $\theta \in [-\mathrm{SMALL\_TH}, \mathrm{SMALL\_TH}]$, SMALL_TH = 0.13.
- **Entanglers**: ring pattern, $\mathrm{CX}(q, (q + 1) \bmod N)$.

Optionally, a deterministic *sign bias* can be derived from a golden-angle template (Sec. 7) without changing any other parameters.

4 REF Sampling, Neighbor Labels, and Keep Probabilities

Run the full op list once with REF rounding and no refraction/CWT to collect $B =$ SHOTS bitstrings. For each edge $(q, (q+1) \bmod N)$, compute the Pearson correlation

$$r_q = \frac{\mathbb{E}\left[(X_q - \bar{X}_q)(X_{q+1} - \bar{X}_{q+1})\right]}{\sigma(X_q)\,\sigma(X_{q+1}) + \varepsilon}, \quad X_q \in \{0, 1\}.$$

Threshold to labels $\ell_q \in \{+1, 0, -1\}$ using (τ_+, τ_-). Map labels to keep probabilities p_q:

$$p_q = \begin{cases} 1.00 & \ell_q = +1\ (\alpha) \\ 0.00 & \ell_q = 0 \\ 0.00 & \ell_q = -1\ (\beta) \end{cases}$$

These p_q drive the per-op mask (Sec. 5).

5 Deterministic Refraction Mask

Given ops and $\{p_q\}$:

- H: always keep.
- R_x/R_y: if $|\theta| < T_{\mathrm{TN}}$, keep with probability p_q (tied to seed); else keep.
- $\mathrm{CX}(c, t)$: if either endpoint is β-like ($p = 0$), skip with probability SKIP_CX_BETA (here 1.0); else keep.

A single PRNG seeded derivation ensures identical masks for BASE+R and MOD.

6 Execution Suite and Metrics

We execute:

1. REF (tighter rounding, no refraction, CWT off) $\to$ reference distribution P_{ref}.
2. BASE (aggressive rounding, no refraction).
3. BASE+R (aggressive rounding, with mask).
4. MOD (aggressive rounding, with mask, CWT on).

Metrics vs REF Let P be the empirical distribution of a variant, P_{ref} the REF distribution.

- **Top-K Total Variation**:

$$\mathrm{TV}_K = \frac{1}{2}\left(\left|1 - \sum_{s\in S} P_{\text{ref}}(s)\right| + \sum_{s\in S} \left|P_{\text{ref}}(s) - P(s)\right| \right),$$

  where S is the set of top-K states by P_{ref} mass.
- L_1**(1q)**: mean absolute difference of single-qubit marginals.
- L_1**(2q)**: mean TV over specified pairs' joint distributions.

Runtimes for each configuration are recorded.

7 Anchors & Differences: the Calculation Layer

This layer produces a small, deterministic signature from the REF labels. It does *not* alter any knobs.

7.1 Golden-Angle Target and Phase Lock

Define a deterministic *edge* template $t_i \in \{-1, 0, +1\}$ for edges $(i, (i{+}1) \bmod N)$:

$$t_i = \mathrm{sign}_\varepsilon\big(\sin(i \cdot \theta_\varphi)\big), \quad \theta_\varphi = 2\pi\left(1 - \frac{1}{\varphi}\right), \quad \varphi = \frac{1+\sqrt{5}}{2},$$

where $\mathrm{sign}_\varepsilon(x) = 0$ if $|x| < \varepsilon$, else $\mathrm{sign}(x)$. In the provided code, ε=0 (no zeros in target).

Let measured labels be $\ell_i \in \{-1, 0, +1\}$. Choose the cyclic shift $k^{\in\{0,\dots,N-1\}}$ maximizing matches:

$$k^{=\arg\max_k \sum_i \mathbf{1}\{t_{i-k}=\ell_i\}.}$$

Work with the aligned target $t'_i = t_{i-k}$.

7.2 Correlation Scores

Map labels to signed values $s_i \in \{-1, 0, +1\}$ (same numerical encoding). Define:

$$C = \frac{1}{N}\sum_{i=0}^{N-1} s_i\, t'_i, \qquad C_w = \frac{\sum_i w_i\, s_i\, t'_i}{\sum_i w_i}, \quad w_i = |r_i|.$$

C measures average agreement; C_w up-weights high-confidence edges.

7.3 Balanced-Ternary Differences and Integer

Per-edge agreement $e_i = s_i t'_i \in \{-1, 0, +1\}$. Map to digits $d_i \in \{-1, 0, +1\}$:

$$d_i = \begin{cases} 0 & e_i = +1 \quad \text{(match)} \\ +1 & e_i = -1 \quad \text{(flip } \alpha \leftrightarrow \beta\text{)} \\ -1 & e_i = 0 \quad \text{(mute: measured 0)} \end{cases}$$

Read least-significant trit at $i{=}0$ to form the balanced-ternary integer

$$D = \sum_{i=0}^{N-1} d_i \, 3^i.$$

For human inspection, we also print the trit string using symbols `-0+` (LSB first).

Why this helps

- k makes the signature rotation-invariant.
- (C, C_w) give scalar health/fidelity checks tied to REF behavior.
- $(d_i)/D$ act as a compact checksum or provenance token derived from *actual* REF-quality dynamics.

8 Reading the Console Output

The script prints:

- Synergy map (`labels`, `corr`, `keep`) and β-like count.
- CALC EMBED CHECK (target vs. labels).
- CALC RESULTS: k, anchor/flip/mute indices/counts, (C, C_w), D, trit string.
- REF top states & variant probabilities.
- Summary metrics (TopKTV, L_1(1q/2q)) and runtimes.

Example (abridged)

```
=== CALC RESULTS (anchors & differences) ===
phase-lock shift k* : 3
anchors : 11 / 20 ; flips : 3 / 20 ; mutes : 6 / 20
C = +0.4000 ; Cw = +0.6299
D = -1214834923
d = -000+-000-+000+--00-
```

9 Performance and Reproducibility Notes

- REF and BASE runs dominate runtime when rotations are heavily biased; the mask (BASE+R/MOD) can be much faster due to pruning.
- Seed pinning ensures identical op lists and mask PRNG; floating-point minutiae may vary across GPUs/drivers.
- For more stable D or per-trit reliability, average across runs and majority-vote (d_i) before forming D.

10 Use Cases

10.1 Run Fingerprint / Provenance

Publish $\{k^{,C_w,D,\text{trit string}}\}$ per run as a rotation-invariant signature. Optionally reduce to D mod M for fixed-width IDs.

10.2 Health / Drift Detection

Track C_w, anchor/flip/mute counts over time; alert on drops in C_w or spikes in mutes (potential rounding/driver drift).

10.3 Adaptive Heuristics (Mask Tuning)

Across runs, mark edges with persistent mutes as low-value pruning candidates; preserve edges with persistent flips (information-bearing).

10.4 Single-Number Bench Comparisons

For CWT sweeps, use ΔC_w at fixed mask as a compact effect size aligned to REF labels.

11 Potential Future Directions

1. **Held-out REF for Mask/Score Decoupling:** Build mask from REF-A, evaluate against REF-B to reduce coupling artifacts.
2. **Windowed Signatures:** Partition edges into W windows; emit $(C_w^{(w)})_{w=1}^{W}$ and windowed ternary strings for higher throughput.
3. **Confidence-Aware Digits:** Weight d_i by smooth functions of $|r_i|$ with rounding to $\{-1, 0, +1\}$ after aggregation across runs.
4. **Alternative Templates:** Use multiple incommensurate templates (e.g., phase-shifted golden sequences) and concatenate signatures.
5. **CRC/Hash Layer:** Compute a small CRC over the ternary string for rapid integrity checks.
6. **Mask Learning Loop:** Close a feedback loop: use signature statistics to re-parameterize keep/skip policies over time.
7. **Robust Statistics:** Replace Pearson with rank-based correlation in the labeler for heavy-tailed shot noise.
8. **Cross-HW Validation:** Systematically assess signature sensitivity across GPUs/drivers and rounding modes.

12 How to Run

```
python synergy_refraction_qrack_stable.py
```

Uses GPU device 0 by default; TN threshold pinned to avoid TN in the calibration/REF path. REF uses rounding 0.020; aggressive rounding comes from the environment (set to 0.60 in the script). All other knobs are unchanged from the code.

Acknowledgments

This document accompanies `synergy_refraction_qrack_stable.py` and explains the synergy-guided refraction map, its calculation layer, and downstream applications.