Spectral Geometry of Vacuum Entanglement: Fermions, Part I

Thomas A. Graham

August 25, 2026

Abstract

The weighted prism Q_4 is used to construct direct vertex–edge spectral reads for the top quark, bottom quark, tau lepton, and neutrino. The vertex and edge Laplacians are factorized through the weighted incidence operator, while the edge response is defined by the holonomy-averaged bond propagator. Finite minimization over simple prism contours gives charged-sector Yukawa residuals of -0.000549%, -0.000407%, and $+0.002049\%$ for t, b, and τ, respectively, and a neutral-sector Yukawa residual of -0.000033%. Masses are obtained from the prism vacuum scale $v = 246.897222\,\text{GeV}$. The common reading surface is fixed independently by the neutral Higgs–gauge mass eigenvalue, whose self-consistent positive root lies at the M_Z scale.

Contents

1 Introduction

The weighted prism Q_4, its vertex and edge operators, and the bond action used below are constructed in Ref. [1], where the gauge couplings, Newton's constant, the electroweak vacuum scale, the string tension and the glueball are obtained from the same spectrum.

Programs that obtain Standard Model flavour parameters from structure rather than from a fit divide into a few families. The spectral action of noncommutative geometry [2, 3] recovers the Standard Model coupled to gravity from a finite spectral triple; there the Yukawa couplings enter as free parameters of the finite Dirac operator. Froggatt–Nielsen constructions [4] and their grand-unified descendants generate hierarchies as powers of a symmetry-breaking ratio and fix Yukawa ratios by group-theoretic factors, leaving the scale of each channel to be matched. Renormalization-group analyses supply the values against which such constructions are matched [5, 6].

The construction below fixes absolute values rather than ratios. The top, bottom, tau and neutrino Yukawa couplings at M_Z are read as single vertex–edge contractions on the weighted prism Q_4, with no continuous parameter adjusted. The prism conventions and the constants $\log\chi$, S_*, ξ_*, c_D, c_I, c_C and the rule fixing Θ are carried unchanged from Ref. [1]; c_T is fixed in Section 3.

2 Weighted prism and factorization

$$Q_4 = \{(x, y, z, \tau) : x, y, z, \tau \in \{0, 1\}\}, \qquad |V| = 16, \qquad |E^+| = 32. \tag{1}$$

$$A = \Omega^{1/2} d_0, \qquad \Omega_{ee} = \begin{cases} \omega, & e \parallel x, y, \\ 1, & e \parallel z, \tau, \end{cases} \qquad \omega = 1 + \frac{5\sqrt{2}}{18} = 1.392837100659. \tag{2}$$

$$L_V = A^\dagger A, \qquad L_E = AA^\dagger, \qquad L_V|v_\lambda\rangle = \lambda|v_\lambda\rangle \Longrightarrow L_E A|v_\lambda\rangle = AL_V|v_\lambda\rangle = \lambda A|v_\lambda\rangle. \tag{3}$$

In Ref. [1] the operators L_V and L_E are written L_ω and L_1; L_E here carries the weighting Ω. Q_4 is a product of four two-point factors, two weighted by ω and two unweighted, so L_V has eigenvalues $\lambda(a, b)$ with multiplicity $\binom{2}{a}\binom{2}{b}$, where a counts weighted and b unweighted factors carrying the alternating mode:

$$\lambda(a, b) = 2\omega a + 2b, \qquad \Sigma_V = \mathrm{Tr}(1 + L_V)^{-1} = \sum_{a,b=0}^{2} \frac{\binom{2}{a}\binom{2}{b}}{1 + 2\omega a + 2b} = 3.823325615627. \tag{4}$$

$$\widehat{H} = \mathrm{diag}[(-1)^\tau], \qquad \langle\psi\rangle = h_0 \begin{pmatrix} 0 \\ v \end{pmatrix}, \qquad h_0 = \frac{1}{\sqrt{2}}, \qquad v = 246.897222 \text{ GeV}. \tag{5}$$

The vacuum scale v is derived in Appendix A. The doublet normalization h_0 and the Higgs-mode weight G_ψ fix the mass of a fermion carried by the prism. Writing the read as a mass rather than a coupling,

$$m_f = h_0\, v\, y_f = h_0 \left(m_P\, \Theta_\psi \frac{G_\psi}{\Sigma_V}\, e^{-1/\xi_*} \right) \Theta_f |\mathcal{A}_f| = \frac{m_P\, \Theta_\psi \Theta_f\, G_\psi}{\sqrt{2}\, \Sigma_V}\, e^{-1/\xi_*}\, |\mathcal{A}_f|, \tag{6}$$

so with $G_\psi = \frac{1}{3}$ and $h_0 = \frac{1}{\sqrt{2}}$ the prefactor is $1/(3\sqrt{2}\,\Sigma_V)$: the $\sqrt{2}$ is the doublet normalization and the 3 is the reciprocal of the Higgs-mode resolvent weight. Neither is fitted. Numerically,

$$m_f = \frac{1.220900 \times 10^{19} \cdot 2.319522830 \times 10^{-16}}{3\sqrt{2}\,(3.823325615627)}\, \Theta|\mathcal{A}_f| = 174.582\, \Theta|\mathcal{A}_f| \text{ GeV}. \tag{7}$$

Calculated masses use this model value only. Reference masses use the independent Fermi-constant normalization

$$v_{\mathrm{ref}} = (\sqrt{2} G_F)^{-1/2} = 246.2196 \text{ GeV}, \qquad m_f^{\mathrm{calc}} = h_0\, v\, y_f^{\mathrm{calc}}, \qquad m_f^{\mathrm{ref}} = h_0\, v_{\mathrm{ref}}\, y_f^{\mathrm{ref}}. \tag{8}$$

3 Bond operator and edge propagator

Write E_x, E_y, E_z, E_τ for the four edge classes of Eq. (1) and $e_i \sim e_j$ when two distinct edges share a vertex. The three edge operators have zero diagonal and off-diagonal entries

$$[I_z]_{ij} = 1 \quad \text{if } e_i \sim e_j, \text{ exactly one of } e_i, e_j \in E_z, \text{ and neither lies in } E_\tau, \tag{9}$$

$$[D_{xy}]_{ij} = \begin{cases} 1, & e_i \sim e_j \text{ and } e_i, e_j \in E_x \cup E_y, \\ 2, & e_i \parallel e_j \text{ opposite in a common } xy \text{ face}, \end{cases} \tag{10}$$

$$[I_\tau]_{ij} = 1 \quad \text{if } e_i \sim e_j \text{ and exactly one of } e_i, e_j \in E_\tau. \tag{11}$$

Row sums on each operator's own edge class are 4 on E_z, 4 on $E_x \cup E_y$ and 6 on E_τ. In the Peierls factor $\bar{x}_i$ and $\bar{y}_i$ are the x and y coordinates of the midpoint of e_i, and $\widehat{H}$ reads the temporal parity of the source vertex.

$\log \chi$ is the Bekenstein–Hawking capacity of a Planck-area bond and S_* the vacuum entanglement per site, so ξ_* is the residual capacity below the bound and $1/\xi_* = 36$ the depth of the descent [1]. Each coefficient is an incidence count in units of ξ_*: c_I the open star of the z-edge, c_C the two diagonal placements per face, c_D the closed star per unit diagonal [1]. A τ-edge meets three spatial edges at each of its two endpoints, so $(I_\tau \mathbf{1})_k = 6$ on a τ-edge and $c_T = 6\xi_*$, the open star of the temporal edge; with $b_s = 0$ the strong sector is unperturbed. Since $(D_{xy}\mathbf{1})_k = 4$ on $E_x \cup E_y$ and 0 on $E_z \cup E_\tau$, the weight of Eq. (2) is $\omega = 1 + 4c_D$ on transverse edges and 1 elsewhere.

$$\begin{gathered}\log \chi = \frac{S_{\rm BH}\ell_P^2}{A} = \frac{1}{4}, \qquad S_* = \frac{2}{9}, \qquad \xi_* = \log\chi - S_* = \frac{1}{4} - \frac{2}{9} = \frac{1}{36}, \\ c_I = 4\xi_* = \frac{1}{9}, \qquad c_C = 2\xi_* = \frac{1}{18}, \qquad c_T = 6\xi_* = \frac{1}{6}, \qquad c_D = \frac{5}{\sqrt{2}}\xi_* = \frac{5}{36\sqrt{2}}, \\ \omega_k = 1 + c_D\big(D_{xy}\mathbf{1}\big)_k, \qquad C = D_{xy}I_z + I_z D_{xy}.\end{gathered} \tag{12}$$

$$M_E(\phi) = \mathbf{1}_{32} + c_I I_z - c_C C + c_T I_\tau(\phi). \tag{13}$$

$$[I_\tau(\phi)]_{ij} = (I_\tau)_{ij} \exp\left[i\phi \frac{\bar{x}_i + \bar{x}_j}{2}(\bar{y}_j - \bar{y}_i)\right], \qquad 0 \le \phi < 4\pi. \tag{14}$$

$$\overline{G}_E = \frac{1}{4\pi}\int_0^{4\pi} M_E(\phi)^{-1}\,\mathrm{d}\phi = \frac{1}{4\pi}\int_0^{4\pi}\sum_{n=1}^{32} \frac{|n(\phi)\rangle\langle n(\phi)|}{\mu_n(\phi)}\,\mathrm{d}\phi, \qquad M_E(\phi)|n(\phi)\rangle = \mu_n(\phi)|n(\phi)\rangle. \tag{15}$$

All numerical amplitudes below use the periodic 4096-point holonomy average. The prism conventions and spectral construction follow Ref. [1].

4 Direct vertex–edge construction

$$|v_f\rangle = \text{the 16-component vector with a single 1 at } v_f, \qquad |c_f\rangle = \sum_{e_i \in E^+(Q_4)} c_{f,i}|e_i\rangle, \qquad c_{f,i} \in \{-1, 0, +1\}. \tag{16}$$

$$\mathcal{A}_f(v_f, c_f) = \langle v_f|\widehat{H}A^\dagger \overline{G}_E|c_f\rangle. \tag{17}$$

Writing $A^\dagger = d_0^{\mathsf{T}}\Omega^{1/2}$, the readout row has entries $[\widehat{H}A^\dagger]_{v,e} = (-1)^{\tau_v}(d_0)_{e,v}\sqrt{\Omega_{ee}}$, which vanish unless e is one of the four edges at v, and equal $\pm\sqrt{\omega}$ on $E_x \cup E_y$ and ± 1 on $E_z \cup E_\tau$. The amplitude is therefore a four-term sum over the edges at the source vertex, and equivalently a signed sum along the contour:

$$\mathcal{A}_f = \sum_{e \ni v_f} (-1)^{\tau_{v_f}}(d_0)_{e,v_f}\sqrt{\Omega_{ee}}\,\big[\overline{G}_E|c_f\rangle\big]_e = \sum_{r=1}^{\ell}\sigma_r\big[\widehat{H}A^\dagger\overline{G}_E\big]_{v_f,\,i_r} = \frac{1}{4\pi}\int_0^{4\pi}\sum_{n=1}^{32}\frac{\langle v_f|\widehat{H}A^\dagger|n(\phi)\rangle\langle n(\phi)|c_f\rangle}{\mu_n(\phi)}\,\mathrm{d}\phi. \tag{18}$$

Channels are distinguished by the right-handed field, since Q and L are shared: u_R and d_R carry colour and charge and are read on an edge, τ_R carries charge without colour and is read at a vertex, ν_R carries neither and is read through the spectrum as a whole. The representation factor is the trace of the identity on the charge the object carries over the trace of the identity on the channels it propagates through [1],

$$\Theta = \frac{\mathrm{Tr}_{\rm charge}(\mathbf{1})}{\mathrm{Tr}_{\rm channel}(\mathbf{1})}, \qquad \Theta_t = \Theta_b = \frac{3}{8}, \qquad \Theta_\tau = \frac{1}{1} = 1, \qquad \Theta_\nu = \frac{1}{2}, \tag{19}$$

the quarks carrying the fundamental 3 through the adjoint 8, the tau the singlet through the singlet, and the neutrino the singlet through a neutral channel. No charge orients the neutrino channel, so its trace is taken to be two. The Yukawa coupling is

$$y_f = \Theta|\mathcal{A}_f|. \tag{20}$$

The neutral coupling carries a spectral determinant D_ω, a product over the whole vertex spectrum where Σ_V is a sum over it, contracted at a single vertex,

$$D_\omega = \det(1 + L_V)^{-1} = \prod_{a,b=0}^{2}\big(1 + 2\omega a + 2b\big)^{-\binom{2}{a}\binom{2}{b}} = 4.473109081297721 \times 10^{-12}, \qquad y_\nu = \Theta D_\omega|\mathcal{A}_\nu|. \tag{21}$$

The contour domain is the set of simple paths on Q_4, counted once up to reversal. There are

$$N_{\text{path}} = 725408 \tag{22}$$

simple path geometries. For an oriented path

$$p = (u_0, u_1, \ldots, u_\ell), \qquad u_{r-1} \sim u_r, \tag{23}$$

let e_{i_r} be the canonical positive edge joining u_{r-1} and u_r, and let

$$\sigma_r = \begin{cases} +1, & u_{r-1} \to u_r \text{ follows } e_{i_r}, \\ -1, & u_{r-1} \to u_r \text{ runs opposite } e_{i_r}, \end{cases} \qquad |c_p\rangle = \sum_{r=1}^{\ell} \sigma_r |e_{i_r}\rangle. \tag{24}$$

The terminal edge is $e_T = e_{i_\ell}$. The sector substrate is

$$e_{T,t}, e_{T,b} \in E_z, \qquad e_{T,\tau}, e_{T,\nu} \in E_x \cup E_y. \tag{25}$$

For each admissible oriented contour, all 16 vertex reads are evaluated. The finite computational minimization gives the following states.

4.1 Top

$$v_t = 1010. \tag{26}$$

$$c_t = e_{19} - e_{30} - e_{23} + e_{24} - e_{10} - e_9 - e_2 + e_1 + e_{13} + e_{17} - e_{28} - e_{20} + e_{21}. \tag{27}$$

$$p_t = 0111 \to 1111 \to 1101 \to 1001 \to 1011 \to 0011 \to 0010 \to 0000 \to 0100 \to 0110 \to 1110 \to 1100 \to 1000 \to 1010. \tag{28}$$

The terminal edge is $e_{21} : 1000 \to 1010 \in E_z$. The amplitude and normalization are

$$\mathcal{A}_t = 2.578652515314844. \tag{29}$$

$$\begin{aligned} y_t &= \Theta \big| \langle v_t | \widehat{H} A^\dagger \overline{G}_E | c_t \rangle \big| \\ &= \frac{3}{8}(2.578652515314844) = 0.9669946932430664. \end{aligned} \tag{30}$$

$$m_t^{\text{calc}} = 168.820544 \text{ GeV}, \qquad m_t^{\text{ref}} = 168.358132 \text{ GeV}, \qquad \delta y_t = -0.000548786\%, \qquad \delta m_t = +0.274660\%. \tag{31}$$

4.2 Bottom

$$v_b = 1111. \tag{32}$$

$$c_b = e_{14} - e_5 + e_6 + e_{10} - e_{24} - e_{22} + e_{21} - e_7 + e_8 + e_{17} + e_{31} - e_{30}. \tag{33}$$

$$p_b = 0100 \to 0101 \to 0001 \to 0011 \to 1011 \to 1001 \to 1000 \to 1010 \to 0010 \to 0110 \to 1110 \to 1111 \to 1101. \tag{34}$$

The terminal edge is $-e_{30} : 1111 \to 1101$, with $e_{30} \in E_z$. The amplitude and normalization are

$$\mathcal{A}_b = -0.04346648976897427. \tag{35}$$

$$\begin{aligned} y_b &= \Theta \big| \langle v_b | \widehat{H} A^\dagger \overline{G}_E | c_b \rangle \big| \\ &= \frac{3}{8}(0.04346648976897427) = 0.01629993366336535. \end{aligned} \tag{36}$$

$$m_b^{\text{calc}} = 2.845686 \text{ GeV}, \qquad m_b^{\text{ref}} = 2.837888 \text{ GeV}, \qquad \delta y_b = -0.000406973\%, \qquad \delta m_b = +0.274802\%. \tag{37}$$

4.3 Tau

$$v_\tau = 0111. \tag{38}$$

$$c_\tau = -e_0 + e_3 + e_5 - e_{14} + e_{12} + e_{28} + e_{31} - e_{30} - e_{23}. \tag{39}$$

$$p_\tau = 1000 \to 0000 \to 0001 \to 0101 \to 0100 \to 1100 \to 1110 \to 1111 \to 1101 \to 1001. \tag{40}$$

The terminal edge is $-e_{23} : 1101 \to 1001$, with $e_{23} \in E_y$. The amplitude is

$$\mathcal{A}_\tau = -0.009938003625759792. \tag{41}$$

$$\begin{aligned} y_\tau &= \Theta \big| \langle v_\tau | \widehat{H} A^\dagger \overline{G}_E | c_\tau \rangle \big| \\ &= 1(0.009938003625759792) = 0.009938003625759792. \end{aligned} \tag{42}$$

$$m_\tau^{\text{calc}} = 1.735004 \text{ GeV}, \qquad m_\tau^{\text{ref}} = 1.730206 \text{ GeV}, \qquad \delta y_\tau = +0.002049002\%, \qquad \delta m_\tau = +0.277265\%. \tag{43}$$

4.4 Neutrino

$$v_\nu = 1001. \tag{44}$$

$$c_\nu = e_2 + e_9 + e_{11} - e_{16} - e_5 + e_4 - e_{22} + e_{20} + e_{28} - e_{25} + e_{26} + e_{27}. \tag{45}$$

$$p_\nu = 0000 \to 0010 \to 0011 \to 0111 \to 0101 \to 0001 \to 1001 \to 1000 \to 1100 \to 1110 \to 1010 \to 1011 \to 1111. \tag{46}$$

The terminal edge is $e_{27} : 1011 \to 1111 \in E_y$. The amplitude and normalization are

$$\mathcal{A}_\nu = -0.12884107671856335. \tag{47}$$

$$\begin{aligned} y_\nu &= \Theta \det(1 + L_V)^{-1} \big|\langle v_\nu | \widehat{H} A^\dagger \overline{G}_E | c_\nu \rangle\big| \\ &= \frac{1}{2}(4.473109081297721 \times 10^{-12})(0.12884107671856335) = 2.8816009515699107 \times 10^{-13}. \end{aligned} \tag{48}$$

$$m_\nu^{\mathrm{calc}} = \frac{246.897222 \times 10^9}{\sqrt{2}}(2.8816009515699107 \times 10^{-13}) = 0.050307767 \text{ eV}. \tag{49}$$

The NuFIT 6.0 normal-ordering reference is

$$m_\nu^{\mathrm{ref}} = \sqrt{2.517 \times 10^{-3} \text{ eV}^2} = 0.050169712 \text{ eV}, \qquad y_\nu^{\mathrm{ref}} = \frac{\sqrt{2}\, m_\nu^{\mathrm{ref}}}{v_{\mathrm{ref}}\, 10^9} = 2.8816019155 \times 10^{-13}. \tag{50}$$

$$\delta y_\nu = -0.0000334512\%, \qquad \delta m_\nu = +0.2751768\%. \tag{51}$$

The neutral reference uses the normal-ordering oscillation scale from NuFIT [9]; the direct beta-decay observable remains independently constrained by KATRIN [10]. The charged-sector reference Yukawa values are the running M_Z values used in Refs. [5, 6, 11]; the neutral reference is normalized with NuFIT as stated above.

5 Minimization combinatorics

The contour domain of Eq. (22) contains $N_{\mathrm{path}} = 725408$ simple path geometries, each carrying two orientations, so 1,450,816 oriented contours. Every oriented contour was evaluated at all sixteen source vertices, and retained for a channel when its terminal edge lies in that channel's substrate, Eq. (25). A state is a source vertex together with a retained oriented contour; single-edge contours are excluded. This gives

$$16(312280 - 8) = 4{,}996{,}352 \text{ states per quark channel}, \qquad 16(537280 - 16) = 8{,}596{,}224 \text{ states per lepton channel}. \tag{52}$$

The scan is exhaustive over this domain rather than a descent, so the four states of Section 4 are the global minima of $|\delta y_f|$ and no unlisted state of either channel gives a smaller residual. Table 1 gives the population of successive residual windows.

$\lvert\delta y_f\rvert$ window	t	b	τ	ν
$< 1\%$	2996	4915	2016	25856
$< 0.1\%$	302	450	207	2672
$< 0.01\%$	35	35	18	289
$< 0.001\%$	1	2	0	37
$< 0.0001\%$	0	0	0	6

Table 1: Counts of states within successive Yukawa-residual windows.

6 Residual summary

f	vertex / contour	terminal edge	ℓ	y_f^{calc}	y_f^{ref}	δy_f	m_f^{calc}	m_f^{ref}	δm_f
t	1010; c_t	e_{21}	13	0.966994693	0.967000000	-0.000549%	168.820544 GeV	168.358132 GeV	$+0.274660\%$
b	1111; c_b	$-e_{30}$	12	0.016299934	0.016300000	-0.000407%	2.845686 GeV	2.837888 GeV	$+0.274802\%$
τ	0111; c_τ	$-e_{23}$	9	0.009938004	0.009937800	$+0.002049\%$	1.735004 GeV	1.730206 GeV	$+0.277265\%$
ν	1001; c_ν	e_{27}	12	2.881601×10^{-13}	2.881602×10^{-13}	-0.000033%	0.0503078 eV	0.0501697 eV	$+0.2752\%$

Table 2: Source vertex, contour, terminal edge and contour length ℓ for each direct prism read, with calculated and reference Yukawa couplings and masses and their fractional residuals.

7 Conclusion

The top, bottom, tau and neutrino Yukawa couplings at M_Z are obtained as single vertex–edge contractions on the weighted prism Q_4, with fractional residuals -0.000549%, -0.000407%, $+0.002049\%$ and -0.000033%. Each source vertex and contour is the global minimum of $|\delta y_f|$ over the enumerated domain of Section 5. The four values fix the normalization of each channel; generation ratios and flavour mixing are not carried by a scalar read and require matrix-valued Yukawa couplings.

A The Higgs mode and the M_Z reading surface

The Higgs field is the temporal mode $\psi = (-1)^\tau$ [1]. With $(1+L_V)^{-1} = \sum_n |n\rangle\langle n|/(\lambda_n+1)$ and $|\psi\rangle = \sum_n c_n|n\rangle$, its propagator weight and spectral weight are

$$G_\psi = \frac{\langle\psi|(1+L_V)^{-1}|\psi\rangle}{\langle\psi|\psi\rangle} = \sum_\lambda \frac{\widehat{c}_\lambda^2}{\lambda+1}, \qquad \widehat{c}_\lambda^2 = \frac{\|P_\lambda\psi\|^2}{|\psi|^2}, \qquad \Pi_\psi = \frac{G_\psi}{\Sigma_V}, \qquad \Sigma_V = \sum_\lambda \frac{m_\lambda}{\lambda+1}, \tag{53}$$

with P_λ the projector onto the eigenspace of λ and m_λ its multiplicity. The weighting Ω acts only on $E_x \cup E_y$ and ψ depends only on τ, so the deformation leaves it untouched and $L_V\psi = L_0\psi = 2\psi$, $\lambda_\psi = 2$. It is one normalized mode, so a single term survives,

$$G_\psi = \frac{1}{\lambda_\psi+1} = \frac{1}{2+1} = \frac{1}{3}, \qquad \Pi_\psi = \frac{G_\psi}{\Sigma_V} = \frac{1/3}{3.823325615627} = 0.087184134140. \tag{54}$$

The mode is a scalar, so the spectral weight enters at $p=1$, and it is a singlet in charge and in channel, so $\Theta = 1/1 = 1$ on the mode. The master equation [1] gives

$$v = M_P\,\Pi_\psi^p\, e^{-1/\xi_*} = 1.220900\times10^{19}\cdot 1\cdot 0.087184134140\cdot 2.319522830\times10^{-16} = 246.897222\ \text{GeV}. \tag{55}$$

The prism vacuum scale and the independently derived electroweak gauge sector determine the neutral Higgs–gauge mass eigenvalue. With

$$\langle\psi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\ v\end{pmatrix}, \qquad v = 246.897222\ \text{GeV}, \tag{56}$$

the neutral mass matrix is

$$\mathcal{M}_0^2 = \frac{v^2}{4}\begin{pmatrix} g_L^2 & -g_Lg_Y \\ -g_Lg_Y & g_Y^2\end{pmatrix}. \tag{57}$$

Its characteristic equation is

$$\det(\mathcal{M}_0^2 - m^2\mathbf{1}_2) = \left(\frac{v^2g_L^2}{4} - m^2\right)\left(\frac{v^2g_Y^2}{4} - m^2\right) - \frac{v^4g_L^2g_Y^2}{16} = m^2\left[m^2 - \frac{v^2}{4}(g_L^2+g_Y^2)\right]. \tag{58}$$

Therefore

$$M_\gamma^2 = 0, \qquad M_Z = \frac{v}{2}\sqrt{g_L^2+g_Y^2}. \tag{59}$$

Because the gauge couplings are evaluated at the mass being determined, the common reading surface is the positive self-consistent root

$$\mu_\star = \frac{v}{2}\sqrt{g_L^2(\mu_\star)+g_Y^2(\mu_\star)}. \tag{60}$$

Both inputs are derived: the vacuum scale from the prism Higgs mode, the gauge couplings from the cube boundary [1],

$$v = 246.897222\ \text{GeV}, \qquad m_P = 1.2209\times10^{19}\ \text{GeV}, \tag{61}$$

$$g_Y^2(m_P) = 0.227406, \qquad g_L^2(m_P) = 0.255268, \qquad g_s^2(m_P) = 0.237350. \tag{62}$$

The couplings evolve as

$$16\pi^2\frac{\mathrm{d}g_i}{\mathrm{d}\ln\mu} = b_ig_i^3, \qquad (b_Y,b_L,b_s) = \left(\frac{41}{6}, -\frac{19}{6}, -7\right), \tag{63}$$

so that

$$\frac{\mathrm{d}g_i^{-2}}{\mathrm{d}\ln\mu} = -2g_i^{-3}\frac{\mathrm{d}g_i}{\mathrm{d}\ln\mu} = -\frac{2b_i}{16\pi^2} \qquad\Longrightarrow\qquad \frac{1}{g_i^2(\mu)} = \frac{1}{g_i^2(m_P)} - \frac{2b_i}{16\pi^2}\ln\frac{\mu}{m_P}. \tag{64}$$

At one loop the root of Eq. (58) is

$$g_L^2 = 0.428044, \qquad g_Y^2 = 0.128048, \qquad \mu_\star^{(1)} = \frac{246.897222}{2}\sqrt{0.428044 + 0.128048} = 92.0576 \text{ GeV}. \tag{65}$$

Retaining no Yukawa term in the scale-defining equation, the two-loop gauge evolution [7] from the same boundary gives

$$g_L^2 = 0.423310, \qquad g_Y^2 = 0.127480, \qquad \mu_\star = \frac{246.897222}{2}\sqrt{0.423310 + 0.127480} = 91.6177 \text{ GeV}, \tag{66}$$

and against the measured value

$$M_Z^{\text{ref}} = 91.1876 \text{ GeV [8]}, \qquad \delta_Z = 100\left(\frac{\mu_\star}{M_Z^{\text{ref}}} - 1\right) = 100\left(\frac{91.6177}{91.1876} - 1\right) = +0.472\%. \tag{67}$$

The two-loop third-family contribution shifts the root by $91.6486 - 91.6177 = 0.0309$ GeV, or 0.034%, and is not used to fix the leading reading surface. Thus the Yukawa reads in the main text are evaluated on the M_Z surface fixed by the Higgs vacuum scale and neutral gauge mass relation, rather than by the fermion residuals.

B Oriented edge library of Q_4

The canonical positive-edge basis used in Eqs. (16)–(24) is listed below. Each edge is specified by its source, target, and prism direction; the orientation fixes the signs appearing in the contour vectors c_f.

edge	$s(e) \to t(e)$	dir.	edge	$s(e) \to t(e)$	dir.
e_0	0000→1000	x	e_{16}	0101→0111	z
e_1	0000→0100	y	e_{17}	0110→1110	x
e_2	0000→0010	z	e_{18}	0110→0111	τ
e_3	0000→0001	τ	e_{19}	0111→1111	x
e_4	0001→1001	x	e_{20}	1000→1100	y
e_5	0001→0101	y	e_{21}	1000→1010	z
e_6	0001→0011	z	e_{22}	1000→1001	τ
e_7	0010→1010	x	e_{23}	1001→1101	y
e_8	0010→0110	y	e_{24}	1001→1011	z
e_9	0010→0011	τ	e_{25}	1010→1110	y
e_{10}	0011→1011	x	e_{26}	1010→1011	τ
e_{11}	0011→0111	y	e_{27}	1011→1111	y
e_{12}	0100→1100	x	e_{28}	1100→1110	z
e_{13}	0100→0110	z	e_{29}	1100→1101	τ
e_{14}	0100→0101	τ	e_{30}	1101→1111	z
e_{15}	0101→1101	x	e_{31}	1110→1111	τ

Table 3: Oriented positive-edge library of the weighted prism Q_4.

C Contour weights and bond response

The amplitude of Eq. (17) is linear in the contour, so for a fixed source vertex it is a signed sum of one weight per prism edge,

$$\mathcal{A}_f = w_{v_f} \cdot c_f, \qquad w_{v_f} = \left(\widehat{H} A^\dagger \overline{G}_E\right)_{v_f,\cdot} \in \mathbb{R}^{32}, \qquad \mathcal{A}_f = \sum_{r=1}^{\ell} \sigma_r\, w_{v_f,\, i_r}. \tag{68}$$

The bond response $\overline{G}_E|c_f\rangle$ carries an amplitude on every one of the thirty-two edges; the readout $\widehat{H}A^\dagger$ is the weighted incidence, so only the four edges incident to v_f enter $\mathcal{A}_f$, with coefficients $\pm\sqrt{\omega}$ on x, y edges and ± 1 on z, τ edges. Each source vertex and contour below is the global minimum of $|\delta y_f|$ over all 1,450,816 oriented contours and all sixteen source vertices.

C.1 Top quark t

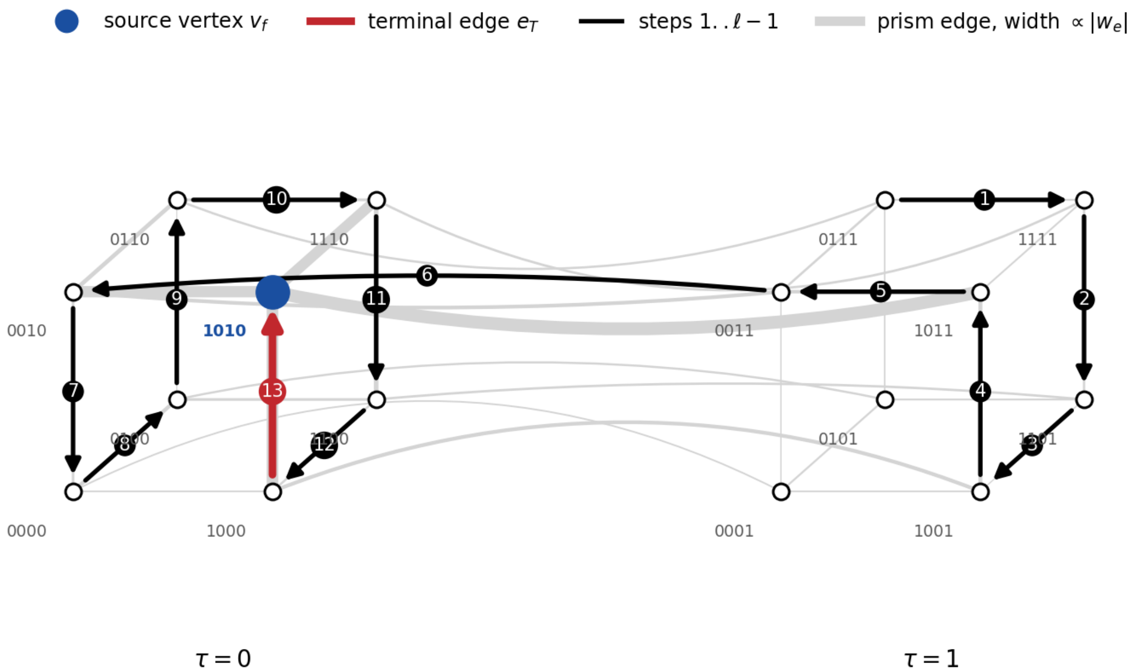

Figure 1: Minimizing contour for t: source vertex 1010, terminal edge e_{21}, contour length 13, $\mathcal{A}_t = +2.578652515315$, $y_t = 0.966994693243$, $\delta y_t = -0.0005488\%$. Grey line width is $|w_e|$ at the source vertex.

e	$s \to t$	dir	amplitude	e	$s \to t$	dir	amplitude
e_0	0000 → 1000	x	+0.1012611	e_{16}	0101 → 0111	z	−0.4086910
$e_1{}^*$	0000 → 0100	y	+1.2762422	$e_{17}{}^*$	0110 → 1110	x	+1.2722963
$e_2{}^*$	0000 → 0010	z	−0.9287049	e_{18}	0110 → 0111	τ	−0.6811871
e_3	0000 → 0001	τ	−0.1199191	$e_{19}{}^*$	0111 → 1111	x	+1.5833606
e_4	0001 → 1001	x	−0.2645977	$e_{20}{}^*$	1000 → 1100	y	−1.2317642
e_5	0001 → 0101	y	+0.1873065	$e_{21}{}^*$	1000 → 1010	z	+1.7605853
e_6	0001 → 0011	z	+0.3480075	e_{22}	1000 → 1001	τ	−0.6024639
e_7	0010 → 1010	x	+0.2040451	$e_{23}{}^*$	1001 → 1101	y	−1.1976230
e_8	0010 → 0110	y	+0.4457093	$e_{24}{}^*$	1001 → 1011	z	+1.7603746
$e_9{}^*$	0010 → 0011	τ	−0.8773361	e_{25}	1010 → 1110	y	−0.0978707
$e_{10}{}^*$	0011 → 1011	x	−1.1618137	e_{26}	1010 → 1011	τ	−0.4617506
e_{11}	0011 → 0111	y	+0.3567736	e_{27}	1011 → 1111	y	−0.0637295
e_{12}	0100 → 1100	x	+0.1636067	$e_{28}{}^*$	1100 → 1110	z	−0.6751964
$e_{13}{}^*$	0100 → 0110	z	+0.8376736	e_{29}	1100 → 1101	τ	+0.2178825
e_{14}	0100 → 0101	τ	−0.4218015	$e_{30}{}^*$	1101 → 1111	z	−1.1984840
e_{15}	0101 → 1101	x	+0.4746710	e_{31}	1110 → 1111	τ	−0.1748693

Table 4: Bond response $\overline{G}_E|c_t\rangle$ on all thirty-two prism edges. Bar length is proportional to |amplitude|, red negative and green positive. Bold rows are the four edges incident to v_t; * marks edges carried by c_t.

edge	readout $(\widehat{H}A^\dagger)_{v,e}$	response	product
e_7	+1.180185	+0.204045088	+0.240810992
e_{21}	+1.000000	+1.760585321	+1.760585321
e_{25}	−1.180185	−0.097870736	+0.115505594
e_{26}	−1.000000	−0.461750608	+0.461750608
		$\mathcal{A}_t =$	+2.578652515

Table 5: Readout of the bond response at v_t.

C.2 Bottom quark b

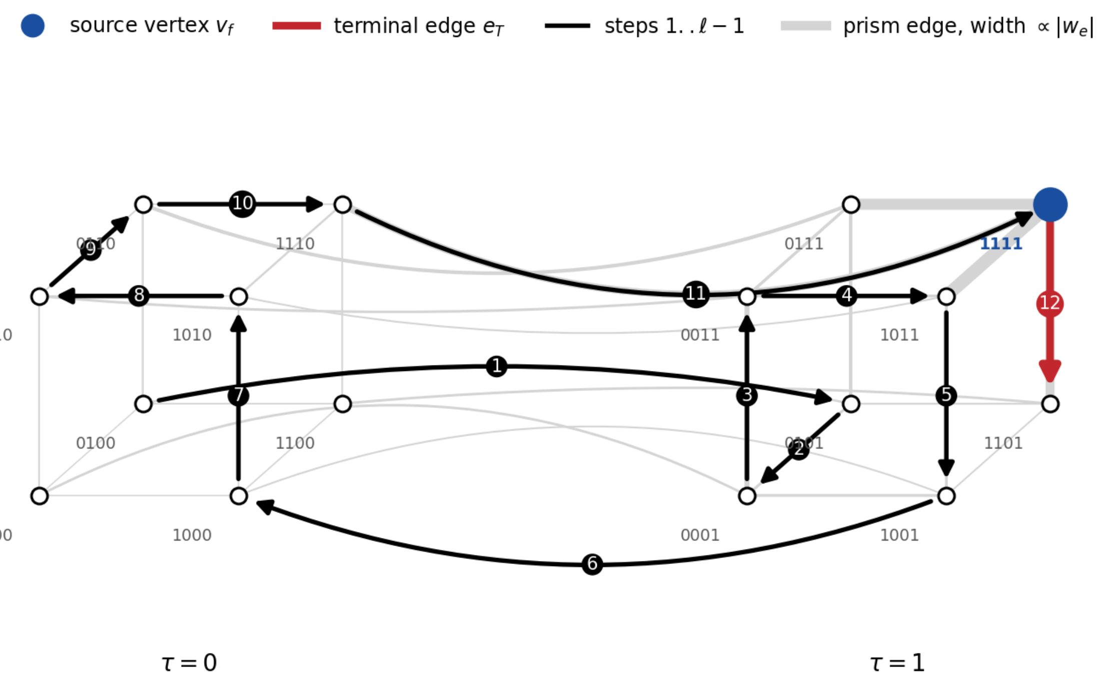

Figure 2: Minimizing contour for b: source vertex 1111, terminal edge e_{30}, contour length 12, $\mathcal{A}_b = -0.043466489769$, $y_b = 0.0162999336634$, $\delta y_b = -0.0004070\%$. Grey line width is $|w_e|$ at the source vertex.

e	$s \to t$	dir	amplitude	e	$s \to t$	dir	amplitude
e_0	0000 → 1000	x	−0.0836080	e_{16}	0101 → 0111	z	+0.2356418
e_1	0000 → 0100	y	+0.0116427	$e_{17}{}^*$	0110 → 1110	x	+1.2170610
e_2	0000 → 0010	z	+0.2122132	e_{18}	0110 → 0111	τ	−0.2768939
e_3	0000 → 0001	τ	+0.0824218	e_{19}	0111 → 1111	x	−0.1045407
e_4	0001 → 1001	x	−0.0359789	e_{20}	1000 → 1100	y	−0.1200030
$e_5{}^*$	0001 → 0101	y	−1.8022667	$e_{21}{}^*$	1000 → 1010	z	+1.7165925
$e_6{}^*$	0001 → 0011	z	+1.2034671	$e_{22}{}^*$	1000 → 1001	τ	−1.1152191
$e_7{}^*$	0010 → 1010	x	−1.1992575	e_{23}	1001 → 1101	y	+0.4199337
$e_8{}^*$	0010 → 0110	y	+1.3567461	$e_{24}{}^*$	1001 → 1011	z	−1.3383074
e_9	0010 → 0011	τ	−0.3273962	e_{25}	1010 → 1110	y	−0.1200030
$e_{10}{}^*$	0011 → 1011	x	+0.8483716	e_{26}	1010 → 1011	τ	−0.0115044
e_{11}	0011 → 0111	y	−0.4571633	e_{27}	1011 → 1111	y	+0.4199337
e_{12}	0100 → 1100	x	+0.0797576	e_{28}	1100 → 1110	z	−0.3051288
e_{13}	0100 → 0110	z	−0.5863814	e_{29}	1100 → 1101	τ	+0.3635216
$e_{14}{}^*$	0100 → 0101	τ	+1.3839083	$e_{30}{}^*$	1101 → 1111	z	−1.5292595
e_{15}	0101 → 1101	x	−0.2418441	$e_{31}{}^*$	1110 → 1111	τ	+1.2005038

Table 6: Bond response $\overline{G}_E|c_b\rangle$ on all thirty-two prism edges. Bar length is proportional to |amplitude|, red negative and green positive. Bold rows are the four edges incident to v_b; * marks edges carried by c_b.

edge	readout $(\widehat{H}A^\dagger)_{v,e}$	response	product
e_{19}	−1.180185	−0.104540717	+0.123377406
e_{27}	−1.180185	+0.419933729	−0.495599571
e_{30}	−1.000000	−1.529259496	+1.529259496
e_{31}	−1.000000	+1.200503821	−1.200503821
		$\mathcal{A}_b =$	−0.043466490

Table 7: Readout of the bond response at v_b.

C.3 Tau lepton τ

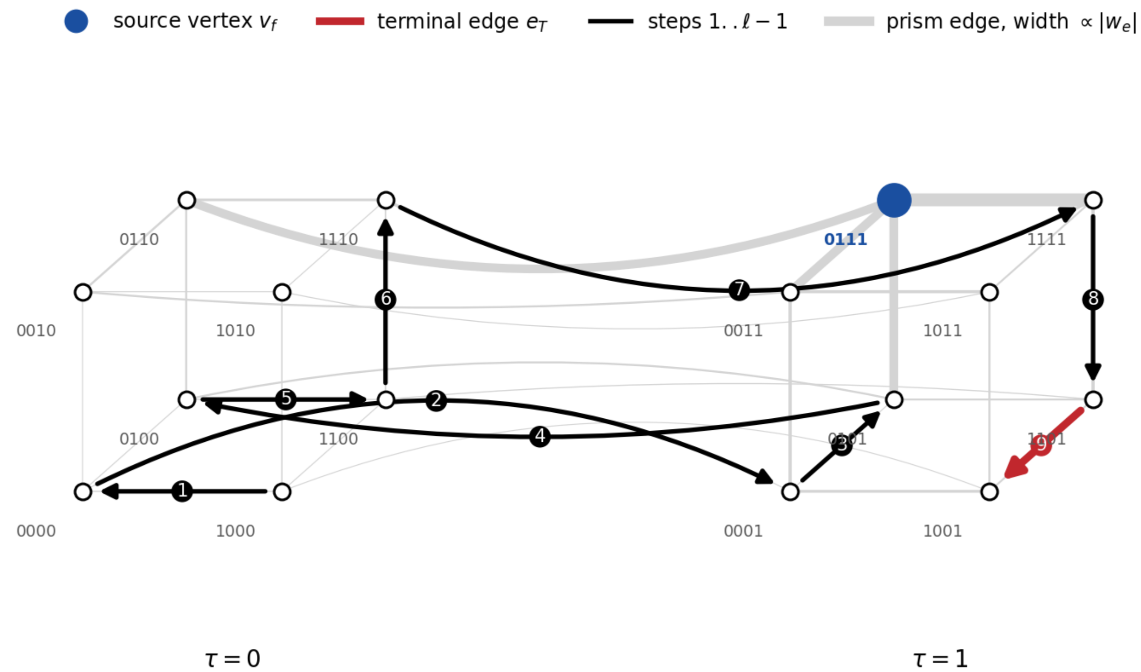

Figure 3: Minimizing contour for τ: source vertex 0111, terminal edge e_{23}, contour length 9, $\mathcal{A}_\tau = -0.009938003626$, $y_\tau = 0.00993800362576$, $\delta y_\tau = +0.0020490\%$. Grey line width is $|w_e|$ at the source vertex.

e	$s \to t$	dir	amplitude	e	$s \to t$	dir	amplitude
$e_0{}^*$	0000 → 1000	x	−1.1347615	e_{16}	0101 → 0111	z	−0.1275564
e_1	0000 → 0100	y	+0.2135408	e_{17}	0110 → 1110	x	−0.2065879
e_2	0000 → 0010	z	−0.0245124	e_{18}	0110 → 0111	τ	+0.0986808
$e_3{}^*$	0000 → 0001	τ	+1.1347034	e_{19}	0111 → 1111	x	−0.1760294
e_4	0001 → 1001	x	−0.4118953	e_{20}	1000 → 1100	y	−0.1508095
$e_5{}^*$	0001 → 0101	y	+0.9364070	e_{21}	1000 → 1010	z	+0.2909288
e_6	0001 → 0011	z	−0.3869993	e_{22}	1000 → 1001	τ	+0.1806019
e_7	0010 → 1010	x	+0.0818269	$e_{23}{}^*$	1001 → 1101	y	−1.1202510
e_8	0010 → 0110	y	+0.1339922	e_{24}	1001 → 1011	z	+0.2361343
e_9	0010 → 0011	τ	+0.0890235	e_{25}	1010 → 1110	y	−0.0838628
e_{10}	0011 → 1011	x	−0.1953069	e_{26}	1010 → 1011	τ	−0.0732486
e_{11}	0011 → 0111	y	−0.1431417	e_{27}	1011 → 1111	y	−0.0533043
$e_{12}{}^*$	0100 → 1100	x	+1.2827788	$e_{28}{}^*$	1100 → 1110	z	+0.6683038
e_{13}	0100 → 0110	z	−0.0727619	e_{29}	1100 → 1101	τ	−0.2044792
$e_{14}{}^*$	0100 → 0101	τ	−1.4242909	$e_{30}{}^*$	1101 → 1111	z	−1.0787984
e_{15}	0101 → 1101	x	+0.3133372	$e_{31}{}^*$	1110 → 1111	τ	+1.2087487

Table 8: Bond response $\overline{G}_E|c_\tau\rangle$ on all thirty-two prism edges. Bar length is proportional to |amplitude|, red negative and green positive. Bold rows are the four edges incident to v_τ; * marks edges carried by c_τ.

edge	readout $(\widehat{H}A^\dagger)_{v,e}$	response	product
e_{11}	−1.180185	−0.143141655	+0.168933663
e_{16}	−1.000000	−0.127556431	+0.127556431
e_{18}	−1.000000	+0.098680841	−0.098680841
e_{19}	+1.180185	−0.176029369	−0.207747256
		$\mathcal{A}_\tau =$	−0.009938004

Table 9: Readout of the bond response at v_τ.

C.4 Neutrino ν

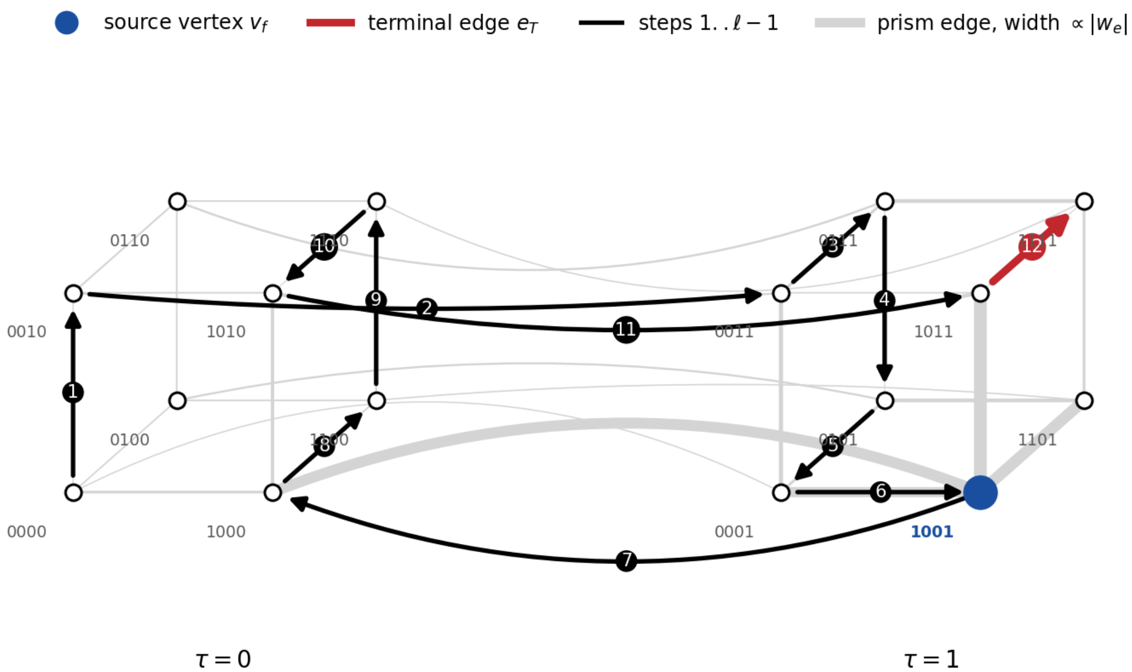

Figure 4: Minimizing contour for ν: source vertex 1001, terminal edge e_{27}, contour length 12, $\mathcal{A}_\nu = -0.128841076719$, $y_\nu = 2.88160095157e-13$, $\delta y_\nu = -0.0000335\%$. Grey line width is $|w_e|$ at the source vertex.

e	$s \to t$	dir	amplitude	e	$s \to t$	dir	amplitude
e_0	0000 → 1000	x	+0.4110862	$e_{16}{}^*$	0101 → 0111	z	−0.7701124
e_1	0000 → 0100	y	+0.1596719	e_{17}	0110 → 1110	x	+0.1776733
$e_2{}^*$	0000 → 0010	z	+0.9988577	e_{18}	0110 → 0111	τ	−0.0871421
e_3	0000 → 0001	τ	−0.3108765	e_{19}	0111 → 1111	x	+0.0970622
$e_4{}^*$	0001 → 1001	x	+1.1766289	$e_{20}{}^*$	1000 → 1100	y	+1.1757819
$e_5{}^*$	0001 → 0101	y	−0.9209393	e_{21}	1000 → 1010	z	+0.0864710
e_6	0001 → 0011	z	+0.0399536	$e_{22}{}^*$	1000 → 1001	τ	−1.3553260
e_7	0010 → 1010	x	−0.1872336	e_{23}	1001 → 1101	y	−0.0586754
e_8	0010 → 0110	y	+0.0319476	e_{24}	1001 → 1011	z	−0.0262793
$e_9{}^*$	0010 → 0011	τ	+0.7644715	$e_{25}{}^*$	1010 → 1110	y	−0.8911648
e_{10}	0011 → 1011	x	−0.4216909	$e_{26}{}^*$	1010 → 1011	τ	+1.1592445
$e_{11}{}^*$	0011 → 0111	y	+0.9513365	$e_{27}{}^*$	1011 → 1111	y	+0.8743779
e_{12}	0100 → 1100	x	+0.1229415	$e_{28}{}^*$	1100 → 1110	z	+1.1271453
e_{13}	0100 → 0110	z	+0.0349456	e_{29}	1100 → 1101	τ	−0.2615441
e_{14}	0100 → 0101	τ	+0.2218604	e_{30}	1101 → 1111	z	+0.1682412
e_{15}	0101 → 1101	x	+0.0423303	e_{31}	1110 → 1111	τ	−0.2809327

Table 10: Bond response $\overline{G}_E|c_\nu\rangle$ on all thirty-two prism edges. Bar length is proportional to |amplitude|, red negative and green positive. Bold rows are the four edges incident to v_ν; * marks edges carried by c_ν.

edge	readout $(\widehat{H}A^\dagger)_{v,e}$	response	product
e_4	−1.180185	+1.176628858	−1.388639961
e_{22}	−1.000000	−1.355326001	+1.355326001
e_{23}	+1.180185	−0.058675397	−0.069247835
e_{24}	+1.000000	−0.026279281	−0.026279281
		$\mathcal{A}_\nu =$	−0.128841077

Table 11: Readout of the bond response at v_ν.

Acknowledgments

Anthropic's Claude (Opus versions 4.6–5) and OpenAI's ChatGPT (GPT-5.6) were used for computational combinatorics/algebraic derivations, Python scripting, and manuscript editing carried out interactively, and PocketCAS for iOS and NetworkX with NumPy in Google Colab to independently verify the matrix calculations. Theoretical direction was provided by the author.

References

[1] T. A. Graham, *Spectral Geometry of Vacuum Entanglement on the Cube and Prism*, Zenodo (2026), doi:10.5281/zenodo.21815012.

[2] A. H. Chamseddine and A. Connes, *The Spectral Action Principle*, Commun. Math. Phys. **186**, 731 (1997), doi:10.1007/s002200050126.

[3] A. H. Chamseddine, A. Connes, and M. Marcolli, *Gravity and the Standard Model with Neutrino Mixing*, Adv. Theor. Math. Phys. **11**, 991 (2007), doi:10.4310/ATMP.2007.v11.n6.a3.

[4] C. D. Froggatt and H. B. Nielsen, *Hierarchy of Quark Masses, Cabibbo Angles and CP Violation*, Nucl. Phys. B **147**, 277 (1979), doi:10.1016/0550-3213(79)90316-X.

[5] S. Antusch, K. Hinze, and S. Saad, *Updated running quark and lepton parameters at various scales*, Phys. Rev. D **113**, 095011 (2026), doi:10.1103/fdcc-ycph.

[6] G.-y. Huang and S. Zhou, *Precise Values of Running Quark and Lepton Masses in the Standard Model*, Phys. Rev. D **103**, 016010 (2021), doi:10.1103/PhysRevD.103.016010.

[7] D. R. T. Jones, *The two-loop beta function for a $G_1 \times G_2$ gauge theory*, Phys. Rev. D **25**, 581 (1982), doi:10.1103/PhysRevD.25.581.

[8] ALEPH, DELPHI, L3, OPAL, SLD Collaborations, LEP Electroweak Working Group, SLD Electroweak and Heavy Flavour Groups, *Precision electroweak measurements on the Z resonance*, Phys. Rept. **427**, 257 (2006), doi:10.1016/j.physrep.2005.12.006.

[9] I. Esteban, M. C. Gonzalez-Garcia, M. Maltoni, I. Martinez-Soler, J. P. Pinheiro, and T. Schwetz, *NuFit-6.0: Updated Global Analysis of Three-Flavor Neutrino Oscillations*, JHEP **12** (2024) 216, doi:10.1007/JHEP12(2024)216.

[10] M. Aker et al. (KATRIN Collaboration), *Direct Neutrino-Mass Measurement Based on 259 Days of KATRIN Data*, Science **388**, 180 (2025), doi:10.1126/science.adq9592.

[11] S. Navas et al. (Particle Data Group), *Review of Particle Physics*, Phys. Rev. D **110**, 030001 (2024), doi:10.1103/PhysRevD.110.030001.

D The edge operators

Rows and columns are indexed by the oriented edge library of Appendix B. Zero entries are shown as a dot. The three operators have zero diagonal; row sums are 4, 4 and 6 respectively, and $C = D_{xy} I_z + I_z D_{xy}$ has row sum 16.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.
1	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	1	1	.	.	.	.	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.
5	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	.	.	1	1	.	.	.	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.
8	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.
11	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
12	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.
13	.	1	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.
14	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.
16	.	.	.	.	.	1	.	.	.	.	.	1	.	.	.	1	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
19	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.	.
21	1	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	1	.	.	.	.	.	.
22	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	1	.
24	.	.	.	.	1	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.
25	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.	.
26	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
27	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	1	.
28	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	1	.	.	1	.	.	.	.	1	.	.	.	.	.	.
29	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
30	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	1	.	.	.	1	.	.	.	.
31	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Table 12: The transverse-to-z contact operator I_z, Eq. (10).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	.	1	.	.	.	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.
1	1	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	.	.	.	.
2	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.
5	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	.
6	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	1	.	.	.	.	.	.
8	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	2	.	.	.	.	.	.
9	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	2	.	.	.	.	.	.	.	1	.	.	.	.
11	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	2	.	.	.	.
12	2	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
14	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	.	.	.	.	2	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
17	.	.	.	.	.	.	.	2	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
19	.	.	.	.	.	.	.	.	.	.	2	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.
20	1	2	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
22	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
23	.	.	.	.	1	2	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
25	.	.	.	.	.	.	.	1	2	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.
26	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
27	.	.	.	.	.	.	.	.	.	.	1	2	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.
28	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
29	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
30	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
31	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Table 13: The transverse plaquette operator D_{xy}, Eq. (11).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.
1	.	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	.	.	.	1	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	1	1	1	.	1	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.
5	.	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	.	1	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.
8	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	1	.	.	.	1	1	1	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.
11	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.
12	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.
13	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.
14	.	1	.	.	.	1	.	.	.	.	.	.	1	1	.	1	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	1
18	.	.	.	.	.	.	.	.	1	.	.	1	.	1	.	.	1	1	.	1	.	.	.	.	.	.	.	.	.	.	.	.
19	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	1
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.	.
22	1	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	1	.	1	1	.	.	.	.	.	.	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	1	.	.	.	.	.
25	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	1
26	.	.	.	.	.	.	.	1	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	1	1	.	1	.	.	.	.
27	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	1
28	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	1
29	.	.	.	.	.	.	.	.	.	.	.	.	1	.	.	1	.	.	.	.	1	.	.	1	.	.	.	.	1	.	1	.
30	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	1
31	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	1	.	1	.	.	.	.	.	1	.	1	1	.	1	.

Table 14: The temporal star operator I_τ, Eq. (12).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	.	.	1	.	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	.	3	.	.	.
1	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	.	3	.	.	.
2	1	1	.	.	.	.	.	1	1	.	.	.	3	.	.	.	.	3	.	.	3	.	.	.	.	3	.	.	.	.	.	.
3	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	3	.
5	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	3	.
6	.	.	.	.	1	1	.	.	.	.	1	1	.	.	.	3	.	.	.	3	.	.	.	3	.	.	.	3	.	.	.	.
7	.	.	1	.	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	.	3	.	.	.
8	.	.	1	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	.	3	.	.	.
9	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	3	.
11	.	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	3	.
12	.	.	3	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	.	1	.	.	.
13	3	1	.	.	.	.	.	3	1	.	.	.	1	.	.	.	.	1	.	.	3	.	.	.	.	3	.	.	.	.	.	.
14	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
15	.	.	.	.	.	.	3	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	1	.
16	.	.	.	.	3	1	.	.	.	.	3	1	.	.	.	1	.	.	.	1	.	.	.	3	.	.	.	3	.	.	.	.
17	.	.	3	.	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	.	1	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
19	.	.	.	.	.	.	3	.	.	.	.	.	.	.	.	.	1	.	.	.	.	.	.	.	3	.	.	.	.	.	1	.
20	.	.	3	.	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.	.
21	1	3	.	.	.	.	.	1	3	.	.	.	3	.	.	.	.	3	.	.	1	.	.	.	.	1	.	.	.	.	.	.
22	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
23	.	.	.	.	.	.	3	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	1	.
24	.	.	.	.	1	3	.	.	.	.	1	3	.	.	.	3	.	.	.	3	.	.	.	1	.	.	.	1	.	.	.	.
25	.	.	3	.	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	.	1	.	.	.
26	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
27	.	.	.	.	.	.	3	.	.	.	.	.	.	.	.	.	3	.	.	.	.	.	.	.	1	.	.	.	.	.	1	.
28	3	3	.	.	.	.	.	3	3	.	.	.	1	.	.	.	.	1	.	.	1	.	.	.	.	1	.	.	.	.	.	.
29	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
30	.	.	.	.	3	3	.	.	.	.	3	3	.	.	.	1	.	.	.	1	.	.	.	1	.	.	.	1	.	.	.	.
31	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Table 15: The anticommutator $C = D_{xy}I_z + I_zD_{xy}$.

e	$\bar{x}$	$\bar{y}$	Ω_{ee}	e	$\bar{x}$	$\bar{y}$	Ω_{ee}
e_0	0.5	0.0	ω	e_{16}	0.0	1.0	1
e_1	0.0	0.5	ω	e_{17}	0.5	1.0	ω
e_2	0.0	0.0	1	e_{18}	0.0	1.0	1
e_3	0.0	0.0	1	e_{19}	0.5	1.0	ω
e_4	0.5	0.0	ω	e_{20}	1.0	0.5	ω
e_5	0.0	0.5	ω	e_{21}	1.0	0.0	1
e_6	0.0	0.0	1	e_{22}	1.0	0.0	1
e_7	0.5	0.0	ω	e_{23}	1.0	0.5	ω
e_8	0.0	0.5	ω	e_{24}	1.0	0.0	1
e_9	0.0	0.0	1	e_{25}	1.0	0.5	ω
e_{10}	0.5	0.0	ω	e_{26}	1.0	0.0	1
e_{11}	0.0	0.5	ω	e_{27}	1.0	0.5	ω
e_{12}	0.5	1.0	ω	e_{28}	1.0	1.0	1
e_{13}	0.0	1.0	1	e_{29}	1.0	1.0	1
e_{14}	0.0	1.0	1	e_{30}	1.0	1.0	1
e_{15}	0.5	1.0	ω	e_{31}	1.0	1.0	1

Table 16: Edge midpoint coordinates $\bar{x}_i$, $\bar{y}_i$ entering the Peierls factor of Eq. (15), and the diagonal weight Ω_{ee} of Eq. (2).

The readout is $\widehat{H} = \mathrm{diag}[(-1)^{\tau_v}]$ on the sixteen vertices in binary order, $+1$ for $\tau = 0$ and -1 for $\tau = 1$.

Computation script

```
"""
fermions_part1_compute.py
Computations for "Spectral Geometry of Vacuum Entanglement: Fermions, Part I".

Builds the weighted prism Q4, the vertex and edge Laplacians, the bond
operator and its holonomy-averaged propagator, and evaluates the four direct
vertex-edge reads. With --scan it repeats the exhaustive contour scan that
selects the source vertices and terminal edges quoted in the text.

    python3 fermions_part1_compute.py # the four reads
    python3 fermions_part1_compute.py --scan # full scan (~minutes)

Requires numpy only.
"""
import sys
import numpy as np

# ------------------------------------------------------------------ geometry
# Vertices of Q4 as (x,y,z,tau) in binary order 0000 ... 1111.
V = [tuple(int(c) for c in f"{n:04b}") for n in range(16)]
IV = {v: i for i, v in enumerate(V)}

# Canonical positive edges: for each vertex in binary order, take directions
# x,y,z,tau in that order wherever the corresponding bit is 0. This fixes the
# edge library e_0 ... e_31 and the signs sigma_r of every contour.
E = []
for v in V:
    for d in range(4):
        if v[d] == 0:
            t = list(v); t[d] = 1
            E.append((v, tuple(t), d))
NE = 32
DIR = np.array([e[2] for e in E])
X, Y, Z, T = 0, 1, 2, 3

# Unweighted incidence d0 : 16 vertices -> 32 edges.
d0 = np.zeros((NE, 16))
```

```
for i, (s, t, d) in enumerate(E):
    d0[i, IV[s]] = -1.0
    d0[i, IV[t]] = +1.0

# --------------------------------------------------------------- weighting
LOG_CHI = 1.0 / 4.0 # Bekenstein-Hawking bond capacity
S_STAR = 2.0 / 9.0 # vacuum entanglement per site
XI = LOG_CHI - S_STAR # xi_* = 1/36, residual capacity
C_I = 4.0 * XI # open star of the z-edge, = 1/9
C_C = 2.0 * XI # two diagonal placements/face, = 1/18
C_T = 6.0 * XI # open star of the temporal edge, = 1/6
C_D = (5.0 / np.sqrt(2.0)) * XI # closed star per unit diagonal
OMEGA = 1.0 + 4.0 * C_D # 1 + 5 sqrt2 / 18

Om = np.diag([OMEGA if d in (X, Y) else 1.0 for d in DIR])
A = np.sqrt(Om) @ d0 # weighted incidence
L_V = A.T @ A # vertex Laplacian
L_E = A @ A.T # edge Laplacian

SIGMA_V = np.trace(np.linalg.inv(np.eye(16) + L_V))
D_OMEGA = 1.0 / np.linalg.det(np.eye(16) + L_V)

# ---------------------------------------------------------- bond operators
def _shares_vertex(i, j):
    return len({E[i][0], E[i][1]} & {E[j][0], E[j][1]}) > 0

I_z = np.zeros((NE, NE)) # transverse-to-z contact
D_xy = np.zeros((NE, NE)) # transverse plaquette coupling
I_tau = np.zeros((NE, NE)) # temporal star

for i in range(NE):
    for j in range(NE):
        if i == j:
            continue
        # I_z : exactly one of the two edges is a z edge, neither is temporal,
        # and they meet at a vertex. Row sum 4 on E_z.
        if (DIR[i] == Z) != (DIR[j] == Z) and DIR[i] != T and DIR[j] != T \
           and _shares_vertex(i, j):
            I_z[i, j] = 1.0
        # D_xy : both edges transverse. 1 if they meet at a vertex, 2 for the
        # opposite pair of a common xy face. Row sum 4 on E_x u E_y.
        if DIR[i] in (X, Y) and DIR[j] in (X, Y):
            if _shares_vertex(i, j):
                D_xy[i, j] = 1.0
            elif DIR[i] == DIR[j] and E[i][0][Z] == E[j][0][Z] \
                    and E[i][0][T] == E[j][0][T]:
                D_xy[i, j] = 2.0
        # I_tau : exactly one of the two edges is temporal and they meet at a
        # vertex. Row sum 6 on E_tau, the open star of a temporal edge.
        if (DIR[i] == T) != (DIR[j] == T) and _shares_vertex(i, j):
            I_tau[i, j] = 1.0

C_OP = D_xy @ I_z + I_z @ D_xy

# Edge midpoint coordinates, used for the Peierls phase.
MID = np.array([[(E[i][0][k] + E[i][1][k]) / 2.0 for k in range(4)]
                for i in range(NE)])
PHASE = np.zeros((NE, NE))
for i in range(NE):
    for j in range(NE):
        if I_tau[i, j]:
            PHASE[i, j] = (MID[i, X] + MID[j, X]) / 2.0 * (MID[j, Y] - MID[i, Y])
```

```
def bond_operator(phi):
    """M_E(phi) of Eq. (10)."""
    return (np.eye(NE) + C_I * I_z - C_C * C_OP
            + C_T * (I_tau * np.exp(1j * phi * PHASE)))


def edge_propagator(n=4096):
    """Gbar_E: the holonomy average of M_E^-1 over 0 <= phi < 4 pi."""
    S = np.zeros((NE, NE), dtype=complex)
    for k in range(n):
        S += np.linalg.inv(bond_operator(4 * np.pi * k / n))
    return S / n


H_HAT = np.diag([(-1.0) ** v[T] for v in V]) # temporal parity readout

# -------------------------------------------------------------------- reads
THETA = {'t': 3.0 / 8.0, 'b': 3.0 / 8.0, 'tau': 1.0, 'nu': 0.5}
SUBSTRATE = {'t': {Z}, 'b': {Z}, 'tau': {X, Y}, 'nu': {X, Y}}

V_REF = 246.2196 # (sqrt2 G_F)^-1/2
V_MODEL = 246.897222 # prism vacuum scale

Y_REF = {'t': 0.967,
        'b': 0.0163,
        'tau': 0.0099378,
        'nu': np.sqrt(2.0) * np.sqrt(2.517e-3) / (V_REF * 1e9)}

PATHS = {
    't': [0b0111, 0b1111, 0b1101, 0b1001, 0b1011, 0b0011, 0b0010, 0b0000,
          0b0100, 0b0110, 0b1110, 0b1100, 0b1000, 0b1010],
    'b': [0b0100, 0b0101, 0b0001, 0b0011, 0b1011, 0b1001, 0b1000, 0b1010,
          0b0010, 0b0110, 0b1110, 0b1111, 0b1101],
    'tau': [0b1000, 0b0000, 0b0001, 0b0101, 0b0100, 0b1100, 0b1110, 0b1111,
          0b1101, 0b1001],
    'nu': [0b0000, 0b0010, 0b0011, 0b0111, 0b0101, 0b0001, 0b1001, 0b1000,
          0b1100, 0b1110, 0b1010, 0b1011, 0b1111],
}
SOURCE = {'t': 0b1010, 'b': 0b1111, 'tau': 0b0111, 'nu': 0b1001}

EIDX = {}
for i, (s, t, d) in enumerate(E):
    EIDX[(IV[s], IV[t])] = (i, +1)
    EIDX[(IV[t], IV[s])] = (i, -1)


def contour(path):
    """Signed edge vector c_f of Eq. (20) from a vertex sequence."""
    c = np.zeros(NE)
    last = None
    for a, b in zip(path[:-1], path[1:]):
        i, sg = EIDX[(a, b)]
        c[i] += sg
        last = (i, sg)
    return c, last


def yukawa(f, amp):
    """y_f = Theta_f |A_f|, with the spectral determinant on the neutral read."""
    return THETA[f] * abs(amp) * (D_OMEGA if f == 'nu' else 1.0)


def reads():
    G = edge_propagator()
```

```
    readout = np.real(H_HAT @ A.T @ G) # 16 x 32 weights w_v
    rows = []
    for f in ('t', 'b', 'tau', 'nu'):
        c, (eT, sgT) = contour(PATHS[f])
        amp = readout[SOURCE[f]] @ c
        y = yukawa(f, amp)
        m = V_MODEL / np.sqrt(2.0) * y
        m_ref = V_REF / np.sqrt(2.0) * Y_REF[f]
        rows.append((f, "".join(map(str, V[SOURCE[f]])), eT, sgT, len(PATHS[f]) - 1,
                     amp, y, 100 * (y / Y_REF[f] - 1), m, m_ref,
                     100 * (m / m_ref - 1)))
    return readout, rows


# ================================================================ M_Z surface
# Appendix A: the cube-boundary couplings of Ref. [1] descended to the
# self-consistent root of mu = (v/2) sqrt(gL^2(mu) + gY^2(mu)).
M_PLANCK = 1.2209e19
GY2_MP, GL2_MP, GS2_MP = 0.227406, 0.255268, 0.237350
B1 = np.array([41 / 10.0, -19 / 6.0, -7.0]) # g1 GUT-normalised
B2 = np.array([[199 / 50.0, 27 / 10.0, 44 / 5.0], # two-loop gauge, Jones
               [9 / 10.0, 35 / 6.0, 12.0],
               [11 / 10.0, 9 / 2.0, -26.0]])
LOOP = 16 * np.pi ** 2


def _beta(g, two_loop):
    d = g ** 3 / LOOP * B1
    if two_loop:
        d = d + g ** 3 / LOOP ** 2 * (B2 @ g ** 2)
    return d


def descend(mu, two_loop, n=4000):
    """Integrate the gauge couplings from m_P down to mu (RK4 in ln mu)."""
    g = np.array([np.sqrt(5 / 3 * GY2_MP), np.sqrt(GL2_MP), np.sqrt(GS2_MP)])
    h = (np.log(mu) - np.log(M_PLANCK)) / n
    for _ in range(n):
        k1 = _beta(g, two_loop)
        k2 = _beta(g + h / 2 * k1, two_loop)
        k3 = _beta(g + h / 2 * k2, two_loop)
        k4 = _beta(g + h * k3, two_loop)
        g = g + h / 6 * (k1 + 2 * k2 + 2 * k3 + k4)
    return g[1] ** 2, 3 / 5 * g[0] ** 2 # (g_L^2, g_Y^2)


def mz_surface(two_loop=True, n=4000):
    """The positive self-consistent root mu_* and the couplings there."""
    mu = 91.0
    for _ in range(30):
        gL2, gY2 = descend(mu, two_loop, n)
        mu = V_MODEL / 2 * np.sqrt(gL2 + gY2)
    return mu, gL2, gY2


# ---------------------------------------------------------------------- scan
def scan():
    """Exhaustive scan: every simple contour, both orientations, all 16 sources."""
    readout, _ = reads()
    adj = {i: [] for i in range(16)}
    for i, (s, t, d) in enumerate(E):
        adj[IV[s]].append((IV[t], i, +1.0, d))
        adj[IV[t]].append((IV[s], i, -1.0, d))
    W = np.ascontiguousarray(readout)
```

```
    fs = ('t', 'b', 'tau', 'nu')
    windows = [1e-2, 1e-3, 1e-4, 1e-5, 1e-6]
    counts = {f: [0] * 5 for f in fs}
    best = {f: None for f in fs}
    total = [0]

    def walk(u, seen, acc, depth):
        for w, ei, sg, d in adj[u]:
            if (seen >> w) & 1:
                continue
            acc2 = acc + sg * W[:, ei]
            total[0] += 1
            for f in fs:
                if d not in SUBSTRATE[f]:
                    continue
                yv = THETA[f] * np.abs(acc2) * (D_OMEGA if f == 'nu' else 1.0)
                r = np.abs(yv / Y_REF[f] - 1.0)
                for k, wnd in enumerate(windows):
                    counts[f][k] += int((r < wnd).sum())
                vi = int(np.argmin(r))
                if best[f] is None or r[vi] < best[f][0]:
                    best[f] = (r[vi], float(yv[vi]), vi, ei, depth + 1)
            walk(w, seen | (1 << w), acc2, depth + 1)

    for s in range(16):
        walk(s, 1 << s, np.zeros(16), 0)
    return total[0], counts, best


# ------------------------------------------------------------------- main
if __name__ == "__main__":
    print(f"omega = {OMEGA:.12f}")
    print(f"Sigma_V = {SIGMA_V:.12f}")
    print(f"D_omega = {D_OMEGA:.16e}")
    print(f"c_I,c_C,c_T {C_I:.9f} {C_C:.9f} {C_T:.9f}")
    print(f"row sums I_z {I_z.sum(1).max():.0f} on E_z "
          f"D_xy {D_xy.sum(1).max():.0f} on E_x u E_y "
          f"I_tau {I_tau.sum(1).max():.0f} on E_tau")
    _, rows = reads()
    print("\n f vertex e_T l amplitude y_f"
          " dy_f % m_f dm_f %")
    for (f, vs, eT, sgT, ell, amp, y, dy, m, mref, dm) in rows:
        print(f" {f:<4}{vs} {'-' if sgT < 0 else ' '}e{eT:<3} {ell:<3}"
              f"{amp:+.15f} {y:.15g} {dy:+.9f} {m:12.6g} {dm:+.7f}")
    for lab, tl in (("one loop", False), ("two loop", True)):
        mu, gL2, gY2 = mz_surface(tl)
        print(f"\n {lab}: g_L^2={gL2:.9f} g_Y^2={gY2:.9f} mu_*={mu:.8f} GeV"
              f" delta_Z={100*(mu/91.1876-1):+.6f}%")
    if "--scan" in sys.argv:
        n, counts, best = scan()
        print(f"\noriented contours visited: {n}")
        for f in ('t', 'b', 'tau', 'nu'):
            r, y, vi, ei, ell = best[f]
            print(f" {f:<4}global minimum: vertex {''.join(map(str, V[vi]))}"
                  f" e_T e{ei} l={ell} y={y:.15g} |dy|={100*r:.9f}%")
        print("\n window t b tau nu")
        for k, lab in enumerate(["<1%", "<0.1%", "<0.01%", "<0.001%", "<0.0001%"]):
            print(f" {lab:<10}" + "".join(f"{counts[f][k]:>8}"
                                          for f in ('t', 'b', 'tau', 'nu')))
```

Verification script

```
"""
fermions_part1_verify.py
Full mathematical verification of "Spectral Geometry of Vacuum Entanglement:
Fermions, Part I". Every number printed in the paper is recomputed
from the definitions and compared against the printed value.

Standalone: requires numpy only. No other file is needed.

    python3 fermions_part1_verify.py

Exit status 0 if all checks pass.
"""
import sys
from math import comb
import numpy as np

# ================================================================= geometry
V = [tuple(int(c) for c in f"{n:04b}") for n in range(16)]
IV = {v: i for i, v in enumerate(V)}

E = []
for v in V:
    for d in range(4):
        if v[d] == 0:
            t = list(v); t[d] = 1
            E.append((v, tuple(t), d))
NE = 32
DIR = np.array([e[2] for e in E])
X, Y, Z, T = 0, 1, 2, 3

d0 = np.zeros((NE, 16))
for i, (s, t, d) in enumerate(E):
    d0[i, IV[s]] = -1.0
    d0[i, IV[t]] = +1.0

LOG_CHI, S_STAR = 1.0 / 4.0, 2.0 / 9.0
XI = LOG_CHI - S_STAR
C_I, C_C, C_T = 4.0 * XI, 2.0 * XI, 6.0 * XI
C_D = (5.0 / np.sqrt(2.0)) * XI
OMEGA = 1.0 + 4.0 * C_D

Om = np.diag([OMEGA if d in (X, Y) else 1.0 for d in DIR])
A = np.sqrt(Om) @ d0
L_V = A.T @ A
L_E = A @ A.T
SIGMA_V = np.trace(np.linalg.inv(np.eye(16) + L_V))
D_OMEGA = 1.0 / np.linalg.det(np.eye(16) + L_V)


def _meet(i, j):
    return len({E[i][0], E[i][1]} & {E[j][0], E[j][1]}) > 0


I_z = np.zeros((NE, NE))
D_xy = np.zeros((NE, NE))
I_tau = np.zeros((NE, NE))
for i in range(NE):
    for j in range(NE):
        if i == j:
            continue
        if (DIR[i] == Z) != (DIR[j] == Z) and DIR[i] != T and DIR[j] != T \
                and _meet(i, j):
```

```
            I_z[i, j] = 1.0
        if DIR[i] in (X, Y) and DIR[j] in (X, Y):
            if _meet(i, j):
                D_xy[i, j] = 1.0
            elif DIR[i] == DIR[j] and E[i][0][Z] == E[j][0][Z] \
                    and E[i][0][T] == E[j][0][T]:
                D_xy[i, j] = 2.0
        if (DIR[i] == T) != (DIR[j] == T) and _meet(i, j):
            I_tau[i, j] = 1.0
C_OP = D_xy @ I_z + I_z @ D_xy

MID = np.array([[(E[i][0][k] + E[i][1][k]) / 2.0 for k in range(4)]
                for i in range(NE)])
PHASE = np.zeros((NE, NE))
for i in range(NE):
    for j in range(NE):
        if I_tau[i, j]:
            PHASE[i, j] = (MID[i, X] + MID[j, X]) / 2.0 * (MID[j, Y] - MID[i, Y])


def bond_operator(phi):
    return (np.eye(NE) + C_I * I_z - C_C * C_OP
            + C_T * (I_tau * np.exp(1j * phi * PHASE)))


def edge_propagator(n=4096):
    S = np.zeros((NE, NE), dtype=complex)
    for k in range(n):
        S += np.linalg.inv(bond_operator(4 * np.pi * k / n))
    return S / n


H_HAT = np.diag([(-1.0) ** v[T] for v in V])

THETA = {'t': 3.0 / 8.0, 'b': 3.0 / 8.0, 'tau': 1.0, 'nu': 0.5}
SUBSTRATE = {'t': {Z}, 'b': {Z}, 'tau': {X, Y}, 'nu': {X, Y}}
V_REF, V_MODEL = 246.2196, 246.897222
Y_REF = {'t': 0.967, 'b': 0.0163, 'tau': 0.0099378,
         'nu': np.sqrt(2.0) * np.sqrt(2.517e-3) / (V_REF * 1e9)}

PATHS = {
    't': [0b0111, 0b1111, 0b1101, 0b1001, 0b1011, 0b0011, 0b0010, 0b0000,
          0b0100, 0b0110, 0b1110, 0b1100, 0b1000, 0b1010],
    'b': [0b0100, 0b0101, 0b0001, 0b0011, 0b1011, 0b1001, 0b1000, 0b1010,
          0b0010, 0b0110, 0b1110, 0b1111, 0b1101],
    'tau': [0b1000, 0b0000, 0b0001, 0b0101, 0b0100, 0b1100, 0b1110, 0b1111,
          0b1101, 0b1001],
    'nu': [0b0000, 0b0010, 0b0011, 0b0111, 0b0101, 0b0001, 0b1001, 0b1000,
          0b1100, 0b1110, 0b1010, 0b1011, 0b1111],
}
SOURCE = {'t': 0b1010, 'b': 0b1111, 'tau': 0b0111, 'nu': 0b1001}

EIDX = {}
for i, (s, t, d) in enumerate(E):
    EIDX[(IV[s], IV[t])] = (i, +1)
    EIDX[(IV[t], IV[s])] = (i, -1)


def contour(path):
    c = np.zeros(NE)
    last = None
    for a, b in zip(path[:-1], path[1:]):
        i, sg = EIDX[(a, b)]
        c[i] += sg
```

```
        last = (i, sg)
    return c, last


# ================================================================== checks
FAIL, N = [], [0]


def check(label, got, want, tol):
    N[0] += 1
    ok = abs(got - want) <= tol
    print(f"[{'ok ' if ok else 'FAIL'}] {label:<46} {got!r}")
    if not ok:
        FAIL.append((label, got, want))


def check_int(label, got, want):
    N[0] += 1
    ok = got == want
    print(f"[{'ok ' if ok else 'FAIL'}] {label:<46} {got}")
    if not ok:
        FAIL.append((label, got, want))


check_int("|V|", len(V), 16)
check_int("|E+|", len(E), 32)
check("omega", OMEGA, 1.392837100659, 5e-13)
check("xi_* = log chi - S_*", XI, 1.0 / 36.0, 1e-16)
check("c_I", C_I, 1.0 / 9.0, 1e-15)
check("c_C", C_C, 1.0 / 18.0, 1e-15)
check("c_T", C_T, 1.0 / 6.0, 1e-15)

lv = np.sort(np.linalg.eigvalsh(L_V))
le = np.sort(np.linalg.eigvalsh(L_E))
check("spec(L_E) contains spec(L_V)", float(np.abs(lv - le[-16:]).max()), 0.0, 1e-10)

mult = np.sort([2 * OMEGA * a + 2 * b for a in range(3) for b in range(3)
              for _ in range(comb(2, a) * comb(2, b))])
check("lambda(a,b) = 2 omega a + 2 b", float(np.abs(lv - mult).max()), 0.0, 1e-10)
closed = sum(comb(2, a) * comb(2, b) / (1 + 2 * OMEGA * a + 2 * b)
           for a in range(3) for b in range(3))
check("Sigma_V (spectral sum)", closed, 3.823325615627, 5e-13)
check("Sigma_V (trace)", SIGMA_V, 3.823325615627, 5e-13)

psi = np.array([(-1.0) ** v[T] for v in V])
check("L_V psi = 2 psi", float(np.abs(L_V @ psi - 2.0 * psi).max()), 0.0, 1e-12)
check("L_0 psi = 2 psi (undeformed)",
     float(np.abs((d0.T @ d0) @ psi - 2.0 * psi).max()), 0.0, 1e-12)
G_PSI = float(psi @ np.linalg.inv(np.eye(16) + L_V) @ psi / (psi @ psi))
check("G_psi = 1/(lambda_psi+1)", G_PSI, 1.0 / 3.0, 1e-12)
PI_PSI = G_PSI / SIGMA_V
check("Pi_psi", PI_PSI, 0.087184134140, 5e-12)
check("v = m_P Theta_psi Pi_psi exp(-1/xi)",
     1.220900e19 * 1.0 * PI_PSI * np.exp(-36.0), 246.897222, 1e-5)
H0 = 1.0 / np.sqrt(2.0)
K_MASS = 1.220900e19 * np.exp(-36.0) / (3 * np.sqrt(2.0) * SIGMA_V)
check("mass prefactor m_P e^-36 / (3 sqrt2 Sigma_V)", K_MASS, 174.5827, 5e-4)
check("prefactor equals h_0 v", K_MASS - H0 * 246.897222, 0.0, 1e-5)

check("omega = 1 + c_D (D_xy 1)_k on E_x u E_y",
     1.0 + C_D * D_xy.sum(1)[int(np.nonzero(DIR == X)[0][0])], OMEGA, 1e-15)
check_int("I_z row sum on E_z", int(I_z.sum(1)[int(np.nonzero(DIR == Z)[0][0])]), 4)
check_int("D_xy row sum on E_x u E_y",
        int(D_xy.sum(1)[int(np.nonzero(DIR == X)[0][0])]), 4)
```

```
check_int("I_tau row sum on E_tau",
        int(I_tau.sum(1)[int(np.nonzero(DIR == T)[0][0])]), 6)
check_int("I_z off-diagonal entries", int((I_z != 0).sum()), 64)
check_int("D_xy entries", int((D_xy != 0).sum()), 48)
check_int("I_tau entries", int((I_tau != 0).sum()), 96)
check_int("C = {D_xy, I_z} entries", int((C_OP != 0).sum()), 128)
check("C does not vanish", float(np.abs(C_OP).max()), 3.0, 1e-12)
for _nm, _M in (("I_z", I_z), ("D_xy", D_xy), ("I_tau", I_tau), ("C", C_OP)):
    check(f"{_nm} symmetric", float(np.abs(_M - _M.T).max()), 0.0, 1e-12)
    check(f"{_nm} zero diagonal", float(np.abs(np.diag(_M)).max()), 0.0, 1e-12)
check("M_E(phi) hermitian",
      float(np.abs(bond_operator(1.234) - bond_operator(1.234).conj().T).max()),
      0.0, 1e-12)

G = edge_propagator(4096)
check("Gbar_E real", float(np.abs(G.imag).max()), 0.0, 1e-14)
check("Gbar_E converged (4096 vs 8192)",
      float(np.abs(G - edge_propagator(8192)).max()), 0.0, 1e-13)

prod = 1.0
for a in range(3):
    for b in range(3):
        prod *= (1 + 2 * OMEGA * a + 2 * b) ** (comb(2, a) * comb(2, b))
check("D_omega (determinant)", D_OMEGA, 4.473109081297721e-12, 1e-26)
check("D_omega (product over spectrum)", 1.0 / prod, 4.473109081297721e-12, 1e-26)

readout = np.real(H_HAT @ A.T @ G)
PRINTED = {
    't': (+2.578652515314844, 0.9669946932430664, -0.000548786, +0.274660),
    'b': (-0.04346648976897427, 0.01629993366336535, -0.000406973, +0.274802),
    'tau': (-0.009938003625759792, 0.009938003625759792, +0.002049002, +0.277265),
    'nu': (-0.12884107671856335, 2.8816009515699107e-13, -0.000033451, +0.275177),
}
TERMINAL = {'t': (21, +1), 'b': (30, -1), 'tau': (23, -1), 'nu': (27, +1)}
LENGTH = {'t': 13, 'b': 12, 'tau': 9, 'nu': 12}
HA = H_HAT @ A.T
ratio = V_MODEL / V_REF - 1.0

for f in ('t', 'b', 'tau', 'nu'):
    c, (eT, sgT) = contour(PATHS[f])
    amp = readout[SOURCE[f]] @ c
    y = THETA[f] * abs(amp) * (D_OMEGA if f == 'nu' else 1.0)
    dy = 100 * (y / Y_REF[f] - 1)
    m = V_MODEL / np.sqrt(2.0) * y
    dm = 100 * (m / (V_REF / np.sqrt(2.0) * Y_REF[f]) - 1)
    check_int(f"{f}: terminal edge index", eT, TERMINAL[f][0])
    check_int(f"{f}: terminal edge sign", sgT, TERMINAL[f][1])
    check_int(f"{f}: contour length", len(PATHS[f]) - 1, LENGTH[f])
    check_int(f"{f}: terminal edge in substrate", int(DIR[eT]) in SUBSTRATE[f], True)
    check_int(f"{f}: contour is a simple path",
              len(set(PATHS[f])) == len(PATHS[f]) and
              all(bin(a ^ b).count("1") == 1
                  for a, b in zip(PATHS[f][:-1], PATHS[f][1:])), True)
    check(f"{f}: amplitude A_f", amp, PRINTED[f][0], 1e-12)
    check(f"{f}: Yukawa y_f", y, PRINTED[f][1], abs(PRINTED[f][1]) * 1e-12)
    check(f"{f}: residual dy_f (%)", dy, PRINTED[f][2], 1e-6)
    check(f"{f}: residual dm_f (%)", dm, PRINTED[f][3], 1e-5)
    check(f"{f}: (1+dm) = (1+dy)(v/v_ref)",
          (1 + dm / 100) - (1 + dy / 100) * (1 + ratio), 0.0, 1e-9)
    inc = np.nonzero(HA[SOURCE[f]])[0]
    check_int(f"{f}: incident edges at v_f", len(inc), 4)
    resp = np.real(G @ c)
    check(f"{f}: readout closes on A_f",
          float(HA[SOURCE[f]][inc] @ resp[inc]), PRINTED[f][0], 1e-11)
```

```python
# ============================================ Appendix A: the M_Z surface
# The cube-boundary couplings of Ref. [1], integrated down to the
# self-consistent root. Nothing here is hardcoded from the paper.
M_PLANCK = 1.2209e19
GY2_MP, GL2_MP, GS2_MP = 0.227406, 0.255268, 0.237350
B1 = np.array([41 / 10.0, -19 / 6.0, -7.0])
B2 = np.array([[199 / 50.0, 27 / 10.0, 44 / 5.0],
               [9 / 10.0, 35 / 6.0, 12.0],
               [11 / 10.0, 9 / 2.0, -26.0]])
LOOP = 16 * np.pi ** 2


def _beta(g, two_loop):
    d = g ** 3 / LOOP * B1
    if two_loop:
        d = d + g ** 3 / LOOP ** 2 * (B2 @ g ** 2)
    return d


def descend(mu, two_loop, n=4000):
    g = np.array([np.sqrt(5 / 3 * GY2_MP), np.sqrt(GL2_MP), np.sqrt(GS2_MP)])
    h = (np.log(mu) - np.log(M_PLANCK)) / n
    for _ in range(n):
        k1 = _beta(g, two_loop)
        k2 = _beta(g + h / 2 * k1, two_loop)
        k3 = _beta(g + h / 2 * k2, two_loop)
        k4 = _beta(g + h * k3, two_loop)
        g = g + h / 6 * (k1 + 2 * k2 + 2 * k3 + k4)
    return g[1] ** 2, 3 / 5 * g[0] ** 2


def mz_surface(two_loop, n=4000):
    mu = 91.0
    for _ in range(30):
        gL2, gY2 = descend(mu, two_loop, n)
        mu = V_MODEL / 2 * np.sqrt(gL2 + gY2)
    return mu, gL2, gY2


# closed-form one-loop check of the integrator
def _run1(g2_mP, bi, mu):
    return 1.0 / (1.0 / g2_mP - 2 * bi / LOOP * np.log(mu / M_PLANCK))


mu1, gL1, gY1 = mz_surface(False)
check("one-loop g_L^2(mu_*)", gL1, 0.428044, 5e-7)
check("one-loop g_Y^2(mu_*)", gY1, 0.128048, 5e-7)
check("one-loop root mu_*", mu1, 92.0576, 5e-5)
check("integrator matches closed-form one loop",
      gL1 - _run1(GL2_MP, -19 / 6.0, mu1), 0.0, 1e-9)

mu2, gL2, gY2 = mz_surface(True)
check("two-loop g_L^2(mu_*)", gL2, 0.423310, 5e-7)
check("two-loop g_Y^2(mu_*)", gY2, 0.127480, 5e-7)
check("two-loop root mu_*", mu2, 91.6177, 5e-5)
check("mu_* closes on v/2 sqrt(gL^2+gY^2)",
      mu2 - V_MODEL / 2 * np.sqrt(gL2 + gY2), 0.0, 1e-9)
check("delta_Z (%)", 100 * (mu2 / 91.1876 - 1), 0.472, 5e-4)

# integration convergence
mu2b, _, _ = mz_surface(True, n=16000)
check("two-loop root converged (4000 vs 16000 steps)", mu2 - mu2b, 0.0, 1e-6)
```

```
M0sq = V_MODEL ** 2 / 4 * np.array([[gL2, -np.sqrt(gL2 * gY2)],
                                    [-np.sqrt(gL2 * gY2), gY2]])
ev = np.sort(np.linalg.eigvalsh(M0sq))
check("M_gamma = 0", float(ev[0]), 0.0, 1e-10)
check("M_Z eigenvalue = mu_*", float(np.sqrt(ev[1])), mu2, 1e-9)

print()
if FAIL:
    print(f"{len(FAIL)} of {N[0]} checks FAILED:")
    for label, got, want in FAIL:
        print(f" {label}: got {got!r}, expected {want!r}")
    sys.exit(1)
print(f"all {N[0]} checks passed")
```