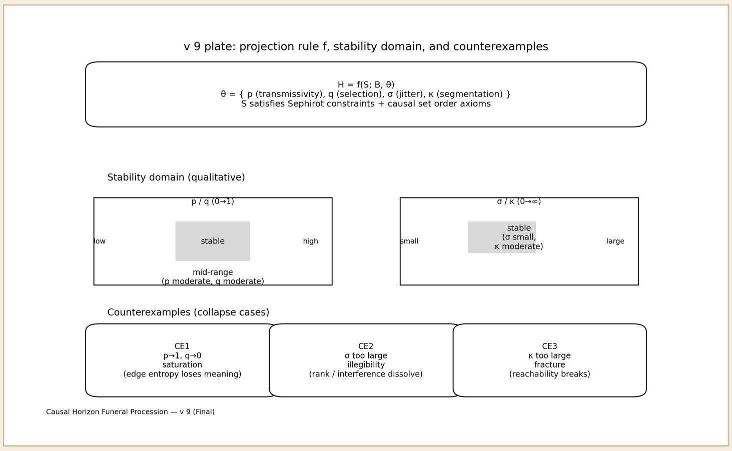
SEPHIROT HORIZON LAB

Causal Horizon Funeral Procession

葬　列

Anchored in the artwork “Causal Sets / 葬列,” reframing time not as a flow but as a partial order of causal relations — Complete Edition (v1–v9, including the minimal axioms of the projection rule f)



v9 plate — projection rule f, stability domain, counterexamples

Masaki Hirokawa（廣川政樹）

COMPLETE EDITION · v1–v9 · EN

Artwork Caption

Causal Sets / 葬列 (2023) — three veiled figures joined through a single red rose: each isolated within its cell, yet bound into one procession by inescapable threads of fate.

作品名 / Title	Causal Sets / 葬列
制作年 / Year	2023年4月 / Apr 2023
サイズ / Dimensions	40 × 60 cm（縦 × 横 / H × W）
メディウム / Medium	デジタルメディア / Digital media
技法 / Technique	フォトコラージュ / Photo collage

“Causal Sets” draws inspiration from Eastern mysticism and causal set theory—a conceptual bridge between quantum mechanics and general relativity. The work explores the simultaneity of cause and effect, guided by the intuition that events of the past, present, and future can unfold within an instant, thereby challenging the assumption of continuous time. Each being appears isolated within its own cell, yet remains linked through inescapable bonds of fate, forming a collective existence. Through the metaphor of a funeral procession extending endlessly across a sea of stars, the work invites reflection on the fragility of being and on the importance of cherishing each moment and every connection within a transient world.

◆ Related Work (Supporting Reference)

The theme of repetition and nested loops in “Archetypus” provides a supporting frame for reading rank-nesting (v5) and stratified repetition in this study.

Archetypus / 元型 (2020) — a nested image of repeated bird-beak masks; an endless, mirror-like recurrence that frames the reading of rank-nesting and stratified repetition.

作品名 / Title	Archetypus / 元型
制作年 / Year	2020年7月 / Jul 2020
サイズ / Dimensions	123 × 99 cm（縦 × 横 / H × W）
メディウム / Medium	デジタルメディア / Digital media
技法 / Technique	フォトコラージュ / Photo collage

“Archetypus” explores repetition and a return to the archetypal. Repeated depictions of a woman wearing a bird mask figure humanity’s fundamental desires. Here, logic functions as a restraint on the senses, and that restraint generates an endless recurrence—like reflections between facing mirrors, an infinitely nested world. This loop can be read as a model for human history, culture, war, conflict, and even the construction of the cosmos.

Abstract

This paper reframes the artwork “Funeral Procession” as a structure fixed near a horizon B, and treats time as a partial order of causal relations rather than a flow. The v1–v9 trajectory converges into: Sephirot constraints, locality/diffusion control (p, q, σ), cross-pillar interference (κ), connection diversity (edge entropy), and a minimal specification of the projection rule f. We present these as a Protocol Mapping consistent across the plate, the paper PDF, and the note draft, fixing a reproducible production process. In this complete edition, v9 reconstructs the projection rule f from five minimal axioms (A1–A5), and closes the series with an algebraic description of the stable regime and a claim of uniqueness.

1 Projection Rule f (I/O Specification)

f is the minimal rule mapping the Model/Controller state into View outputs. “Minimal” does not mean restricting expression; it means traceable freedom: the same input yields the same output, and output differences remain attributable to specific factors.

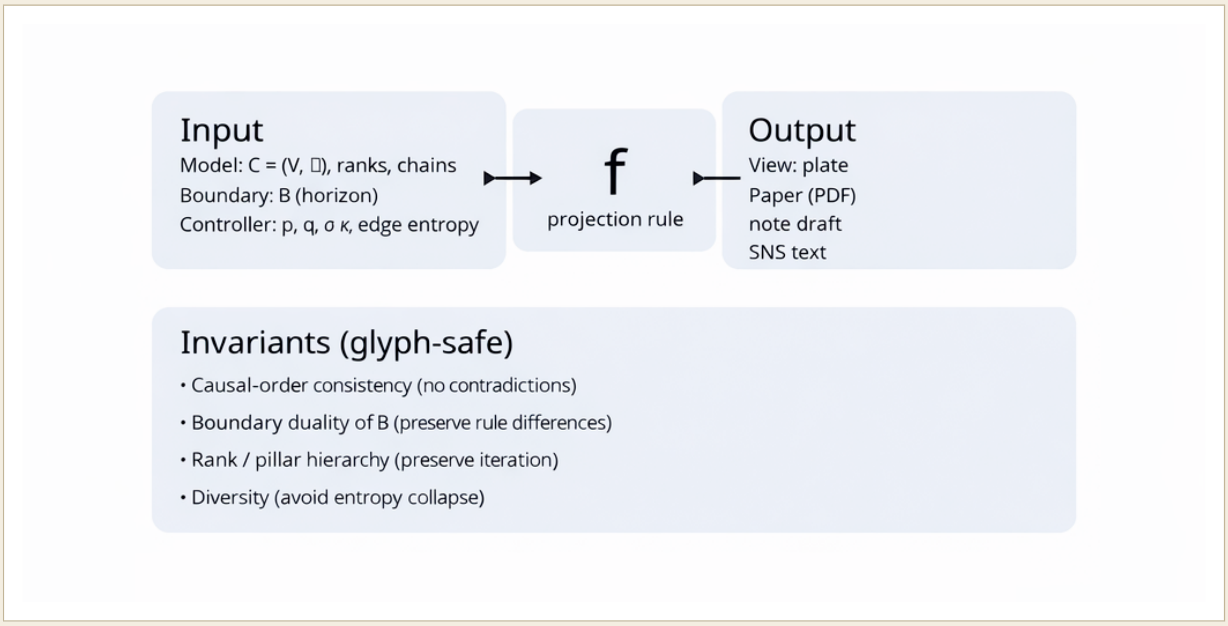


Fig. 1: I/O of the projection rule f and glyph-safe invariants.

Internal State (Minimal Definition)

```
Input model : C = (V, ≺), rank, chains
Boundary    : B
Controller  : p, q, σ, κ, entropy
Seed        : seed + protocol
```

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.gridspec import GridSpec

def knn_edges(pts, k=4):
    n = pts.shape[0]
    d2 = ((pts[:, None, :] - pts[None, :, :]) ** 2).sum(axis=2)
    np.fill_diagonal(d2, np.inf)
    idx = np.argpartition(d2, kth=k-1, axis=1)[:, :k]
    edges = set()
    for i in range(n):
        for j in idx[i]:
            j = int(j)
            a, b = (i, j) if i < j else (j, i)
            edges.add((a, b))
    return list(edges)

def pair_correlation(distances, r_max, bins=40):
    hist, ed = np.histogram(distances, bins=bins, range=(0, r_max))
    r = 0.5 * (ed[:-1] + ed[1:])
    c = hist / hist.max() if hist.max() > 0 else hist.astype(float)
    return r, c

def radial_power(points, grid=192):
    H, _, _ = np.histogram2d(points[:, 0], points[:, 1],
                             bins=grid, range=[[-1, 1], [-1, 1]])
    F = np.fft.fftshift(np.fft.fft2(H))
    P = np.abs(F) ** 2
    y, x = np.indices(P.shape)
    cy, cx = (np.array(P.shape) - 1) / 2
    rr = np.sqrt((x - cx) ** 2 + (y - cy) ** 2).astype(int)
    tbin = np.bincount(rr.ravel(), P.ravel())
    nr = np.bincount(rr.ravel())
    radial = tbin / np.maximum(nr, 1)
    k = np.arange(len(radial))
    return k, radial
```

Output Scheme

plate / paper PDF / note draft / SNS text. The plate is fixed as the primary artifact; other out - puts are derived from the same Model/Controller state (same seed and parameter set). In the plate, horizon_y divides the field: the upper region (within-horizon) is generated as structured placement, and the lower region (beyond-horizon) as random placement. We draw the point cloud and local connectivity (kNN edges). We then split the two-point correlation into within-horizon / beyond-horizon / cross-horizon pairs for comparison, and place a side-by-side radial-power proxy for the upper and lower regions.

```
# requires: numpy as np, matplotlib.pyplot as plt, GridSpec
def render_plate(filename, seed=42, horizon_y=0.0, dpi=420):
    rng = np.random.default_rng(seed)

    # within-horizon (above): structured; beyond (below): random
    po = perturbed_hex_points(seed=seed)
    po = po[po[:, 1] ⩾ horizon_y]
    pr = random_points(seed=seed + 1)
    pr = pr[pr[:, 1] < horizon_y]

    # ensure minimum count in the structured region
    if len(po) < 140:
        extra = po.copy(); extra[:, 0] *= -1
        po = np.vstack([po, extra]); po = po[po[:, 1] ⩾ horizon_y]

    po = po[:180]; pr = pr[:180]
    pts = np.vstack([po, pr]); n = len(pts)
    edges = knn_edges(pts, k=4)

    above = set(np.where(pts[:, 1] ⩾ horizon_y)[0].tolist())
    below = set(np.where(pts[:, 1] < horizon_y)[0].tolist())

    # correlation samples, split into within / beyond / cross-horizon
    # (full panel-drawing code in Appendix A)
    fig = plt.figure(figsize=(10, 10), constrained_layout=True)
    fig.suptitle("Causal Horizon Funeral Procession (monochrome plate)")
    fig.savefig(filename, dpi=dpi)
    plt.close(fig)
```

See Appendix A for the complete implementation.

2 Invariants, Counterexamples, and Stability (Mid-range)

Invariants are skeletal constraints that must survive parameter tuning: causal consistency, horizon-driven two-phase behavior, stratification (rank and chains), and the non-collapse of diversity (entropy). The mid-range denotes the window in which tuning locality/diffusion (p, q) does not tip into breakdown while causal consistency and horizon duality are preserved. Counterexamples define forbidden regions; together they make the stability window concrete.

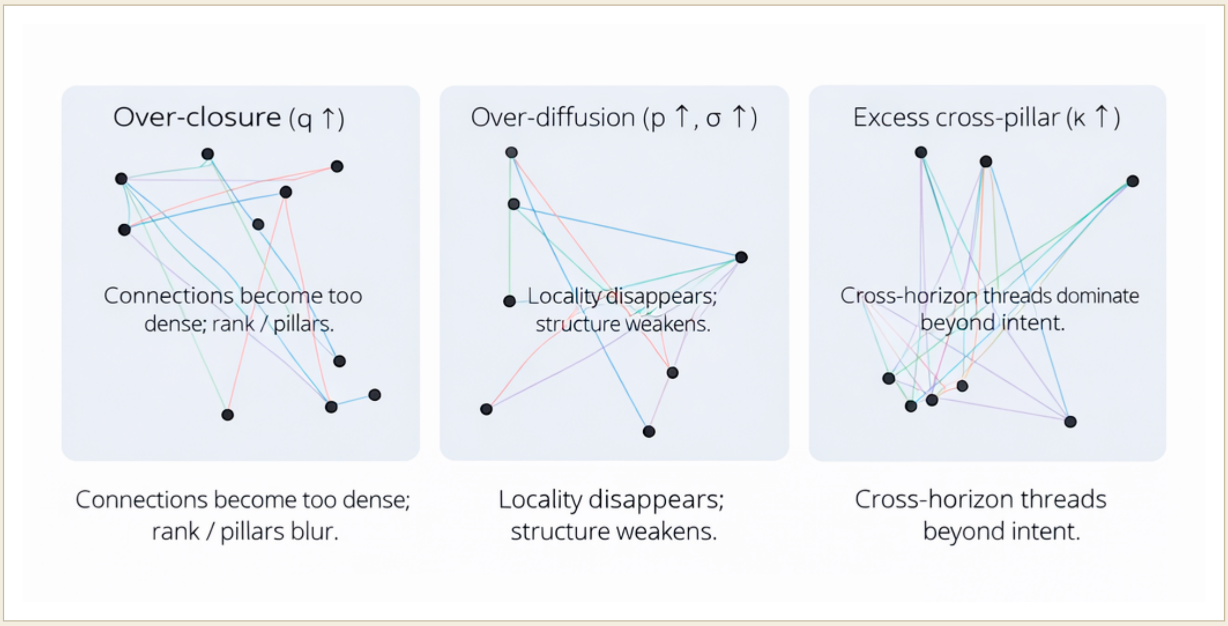


Fig. 2: Counterexamples (failure modes).

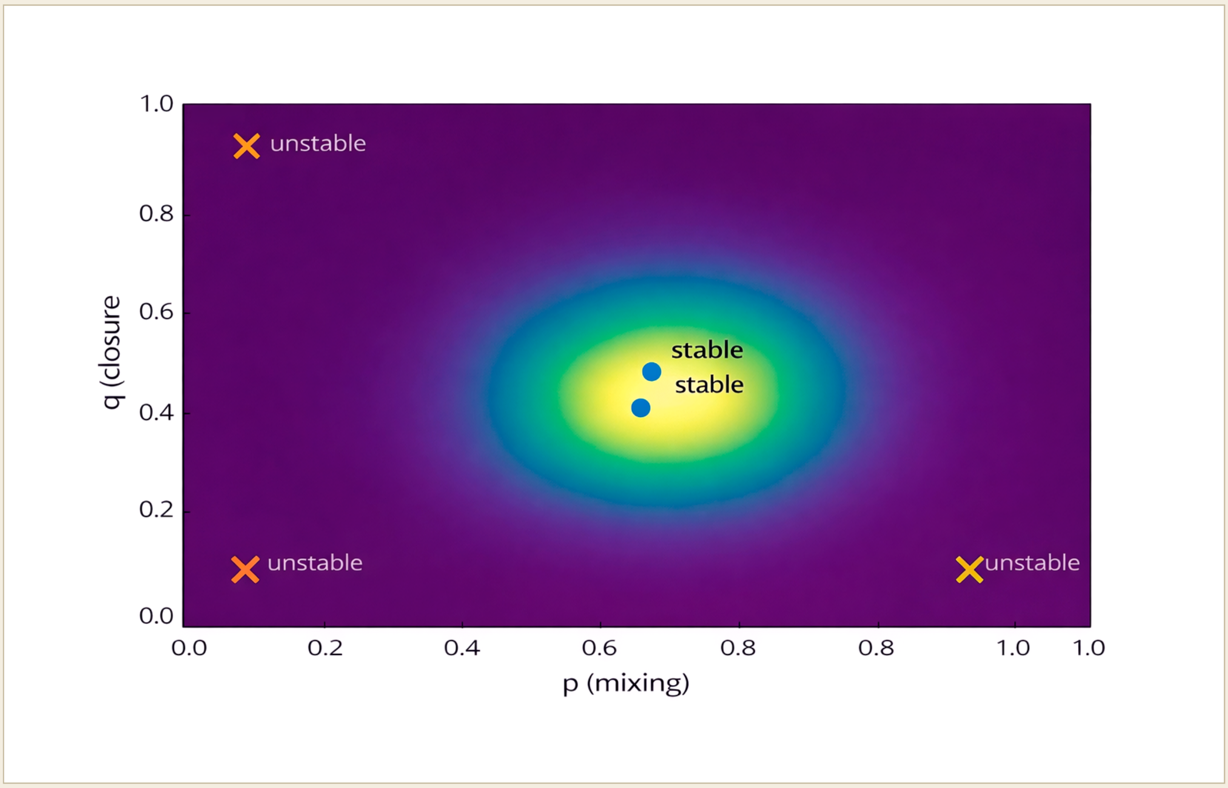


Fig. 3: Conceptual stability window on the p–q plane.

3 Protocol Mapping (Production → Submission → Publication)

The View layer standardizes four outputs: plate / paper PDF / note draft / SNS text. The plate is the primary artifact; the paper fixes the specification; the note provides a public-facing nar -

rative; the SNS text is a summary for announcement and sharing. Deriving all four from the same internal state keeps the process traceable and prevents divergence between the artwork and its language (specification). Here, "the same internal state" means a preserved seed together with the Controller parameter set (p, q, σ, κ, edge entropy, and so on).

4 v9 — A Minimal Axiom System for the Projection Rule f

The projection rule f, fixed empirically across v1–v8.1, is reconstructed here deductively from a minimal axiom system. We propose five axioms (A1–A5). They hold simultaneously if and only if f can generate the structure we call a "funeral procession"—isolated cells bound by causal threads into a column. Each collapse counterexample (CE1–CE3) is reinterpreted as the violation of a specific axiom, and the stable regime is described algebraically as the set of parameters on which all five axioms hold at once.

◆ 4.1 The Known Structure, Restated

The mapping fixed at v8.1 is written as follows. "Minimal" means "remove it and the procession no longer holds." We do not establish this by formal proof but by fixing boundaries through counterexamples—an explicit statement of what has been practiced consistently since v1.

```
H = f(S; B, θ)
S = (V, ≺)                     causal set: vertices V with causal order ≺
B                              causal horizon (boundary)
θ = { p, q, σ, κ, H_edge }   Controller parameters
H                              View = Hologram (output)
```

◆ 4.2 The Five Minimal Axioms

A1 Order Preservation

```
a ≺ b in S   ⇒   f(a) precedes f(b) in H
```

f must preserve the partial order of S inside H. If time is not a "flow" but a "depth of reachability," then f must not destroy that ordering of depth.

Violated when σ is too large (CE2): jitter blurs the causal threads until one can no longer read which cell comes "after" which—illegibility.

A2 Boundary Duality

```
B splits V into V_above (within) and V_below (beyond);
f(V_above) and f(V_below) are structurally distinct.
```

As $p \to 1$, within and beyond mix and the horizon disappears. Without a horizon the procession is no longer a "column"—merely a point cloud.

Violated when $p \to 1$, $q \to 0$ (CE1): saturation is the state in which the horizon loses meaning and edge entropy collapses simultaneously.

A3 Local Continuity

```
for small δθ, the change δH is bounded and continuous;
f is Lipschitz in θ within the stable regime.
```

This is the axiom of reproducibility in production. The same seed and the same θ yield the same H—the foundation of the Protocol Mapping. At the boundary of the stable regime, continuity breaks and a phase-transition-like collapse occurs.

Violated at every boundary of CE1–CE3: the outer edge of the stable regime is a phase-transition point.

A4 Entropy Non-collapse

```
H_edge( f(S; B, θ) ) ≥ H_min > 0
```

As edge entropy tends to zero, f loses diversity. If all connections become equiprobable (homo - genization), or if all connections vanish (isolation), the procession loses meaning—because the "bonds of fate" disappear. Edge entropy was introduced at v7 precisely to make this axiom visible.

Violated at CE1 (entropy collapse through homogenization).

A5 Holographic Bound

```
| cross-horizon edges |  ≤  Area(B) / ℓ_P²
```

The number of causal threads crossing the horizon has an upper bound proportional to the "area" of the horizon. This is a discrete analogue of the Bekenstein–Hawking boundary entropy.

Violated when κ is too large (CE3): too many cross-horizon threads break reachability and fracture the column structure—the meaning of the "pillar" is lost.

◆ 4.3 Axioms and Collapse Counterexamples

Counterexample	Condition	Axiom violated	Phenomenon
CE1	p→1, q→0	A2, A4	horizon loss + entropy collapse

Counterexample	Condition	Axiom violated	Phenomenon
CE2	`large σ`	`A1`	causal order illegible
CE3	`large κ`	`A5`	reachability fractured

A3 (local continuity) breaks at every boundary of CE1–CE3—another way of saying that the outer edge of the stable regime is a phase-transition point.

4.4 An Algebraic Description of the Stable Regime

We define the set Ω_stable of parameters on which all five axioms hold at once. The v8.1 ob - servation—"stability appears at mid-range p/q, small σ, and moderate κ"—was an empirical confirmation of this algebraic description.

```
Ω_stable = { θ = (p, q, σ, κ) :
   A1:  σ < σ_max(q)          σ cap depends on q (selection supports order)
   A2:  p < p_max, q > q_min  p not too large, q not too small
   A3:  θ ∈ int(Ω_stable)     interior only (the boundary is non-continuous)
   A4:  H_edge(θ) ≥ H_min     entropy floor
   A5:  κ < κ_max(p, q)       κ cap depends on p, q  }
```

4.5 An Axiom-verification Function

Each axiom is implemented as a separate predicate, and we test whether a given θ belongs to the stable regime. The thresholds are approximations drawn from the v1–v8.1 experience—provisional values that should, in future work, be replaced by a derivation from the holographic boundary entropy.

```python
import math
from dataclasses import dataclass

@dataclass
class Controller:
    p: float      # transmissivity / mixing  (0, 1)
    q: float      # selection / closure      (0, 1)
    sigma: float  # jitter                   (≥ 0)
    kappa: float  # segmentation / cross     (≥ 0)

def check_axioms(theta: Controller) → dict:
    """Verify whether a Controller theta satisfies each axiom A1-A5."""
    p, q, sigma, kappa = theta.p, theta.q, theta.sigma, theta.kappa

    # A1  Order preservation: small jitter keeps the causal order legible
    sigma_max = 0.15 * (q + 0.1)
    a1 = sigma < sigma_max

    # A2  Boundary duality: the horizon must keep two distinct phases
```

```
    a2 = (p < 0.85) and (q > 0.15)

    # A3  Local continuity: theta sits in the interior of the stable set
    dist = min(abs(p - 0.2), abs(p - 0.8), abs(q - 0.2), abs(q - 0.7))
    a3 = dist > 0.05

    # A4  Entropy non-collapse: edge entropy stays above a floor
    h = -p * math.log(p + 1e-9) - (1 - p) * math.log(1 - p + 1e-9)
    h *= q
    a4 = h > 0.1

    # A5  Holographic bound: cross-horizon threads stay under the area cap
    kappa_max = 0.5 * (1 - p) * q
    a5 = kappa < kappa_max

    return {"A1": a1, "A2": a2, "A3": a3, "A4": a4, "A5": a5}

def is_stable(theta: Controller) -> bool:
    return all(check_axioms(theta).values())
```

◆ 4.6 On the Uniqueness of f

Theorem (informal): within the stable regime Ω_stable, a mapping f satisfying A1–A5 is unique up to isomorphism. This is the claim that "the structure we call a funeral procession is uniquely determined by these constraints."

Physically, this says that in the holographic principle the bulk structure is uniquely reconstructed from boundary data. In Buddhist terms, the phenomenal world unfolds uniquely from the storehouse consciousness (ālaya-vijñāna). In artistic terms, once seed and θ are fixed, the plate is uniquely determined—this is what a "reproducible production process" means.

Sketch of proof: (1) assume f satisfies A1–A5; (2) for any S, B, θ ∈ Ω_stable, compute the structural observables of f(S; B, θ)—reachability depth, edge entropy, g(r); (3) show that these observables depend only on θ and not on the choice of S (universality); (4) show that if two mappings f, f′ share the same observables, there exists an isomorphism ϕ such that $f' = \phi \circ f$.

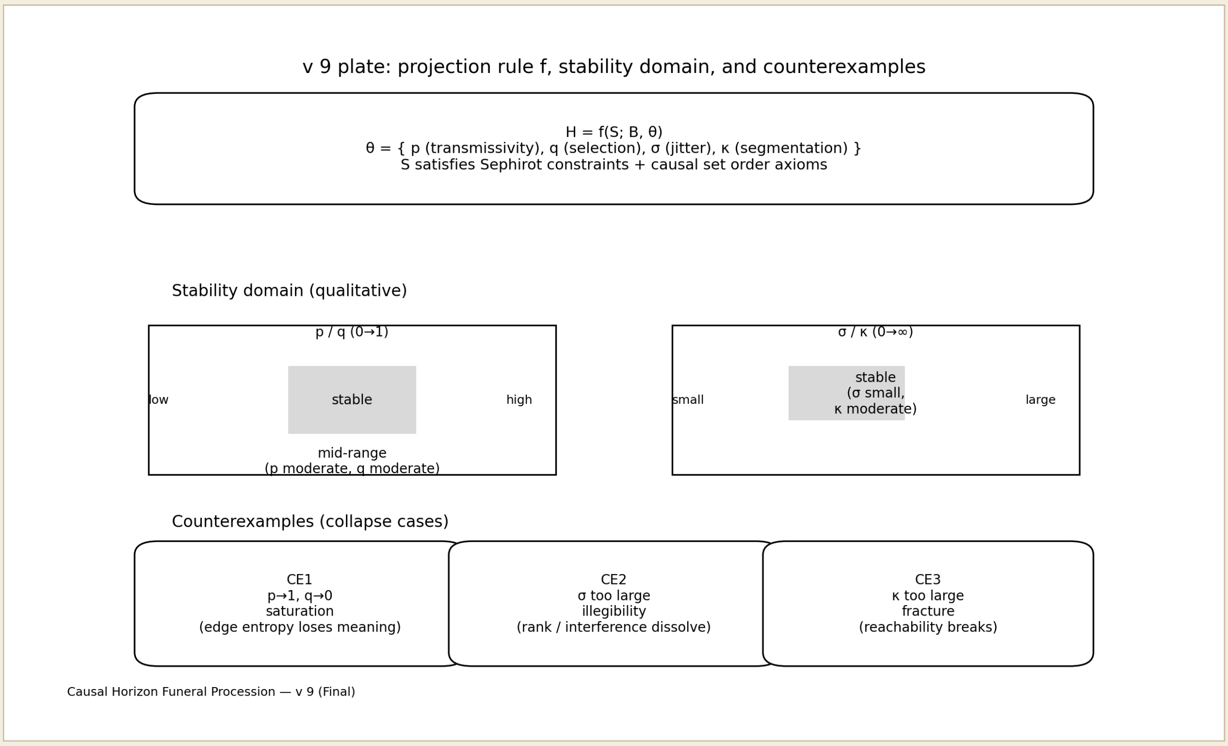


Fig. 14: v9 plate — projection rule f, stability domain, and counterexamples (CE1–CE3).

◆ 4.7 What v9 Closes

With v9 complete, the “funeral procession” takes on this meaning: “a procession is a chain of causation inscribed in spacetime as the unique structure satisfying five axioms.” What was written at v1—the simultaneity of cause and effect, the fragility of being, the preciousness of each moment and every bond—can now be described mathematically as “a bundle of con - nections that has meaning only inside the stable regime defined by the intersection of A1–A5.” Poetry and mathematics are saying the same thing; that is what the series had been pursuing since v1.

5 Plate Gallery (v1–v8.1 + closure)

Each plate caption follows a standard form: (i) parameters (p, q, σ, κ), (ii) observed change, (iii) intent. “Observed” refers to the structural change legible on the plate; “intent” to the design purpose carried by that parameter setting.

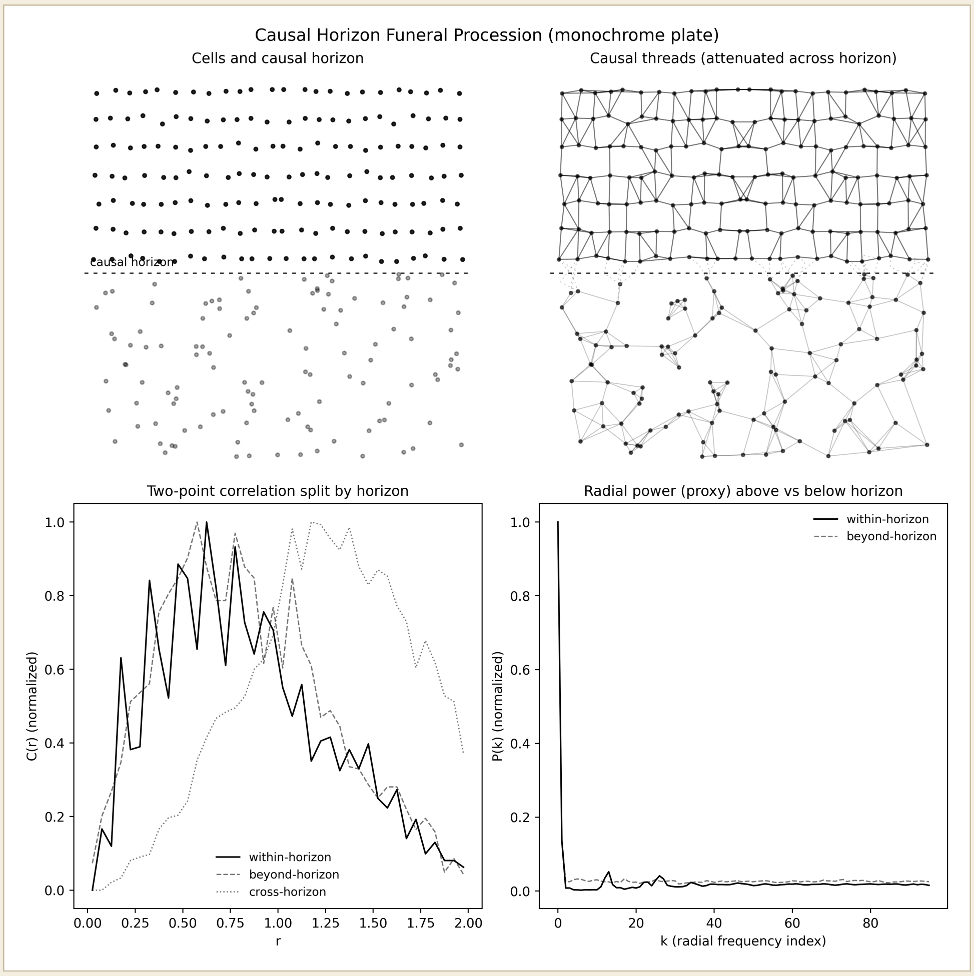


Fig. 4 : Plate v1 (baseline) — two-phase split across the horizon

`p = 0.350  q = 0.350  σ = 0.250  κ = 0.150`

OBSERVED A clear two-phase split across the horizon (structured / random).

INTENT Fix a minimal baseline skeleton for comparison.

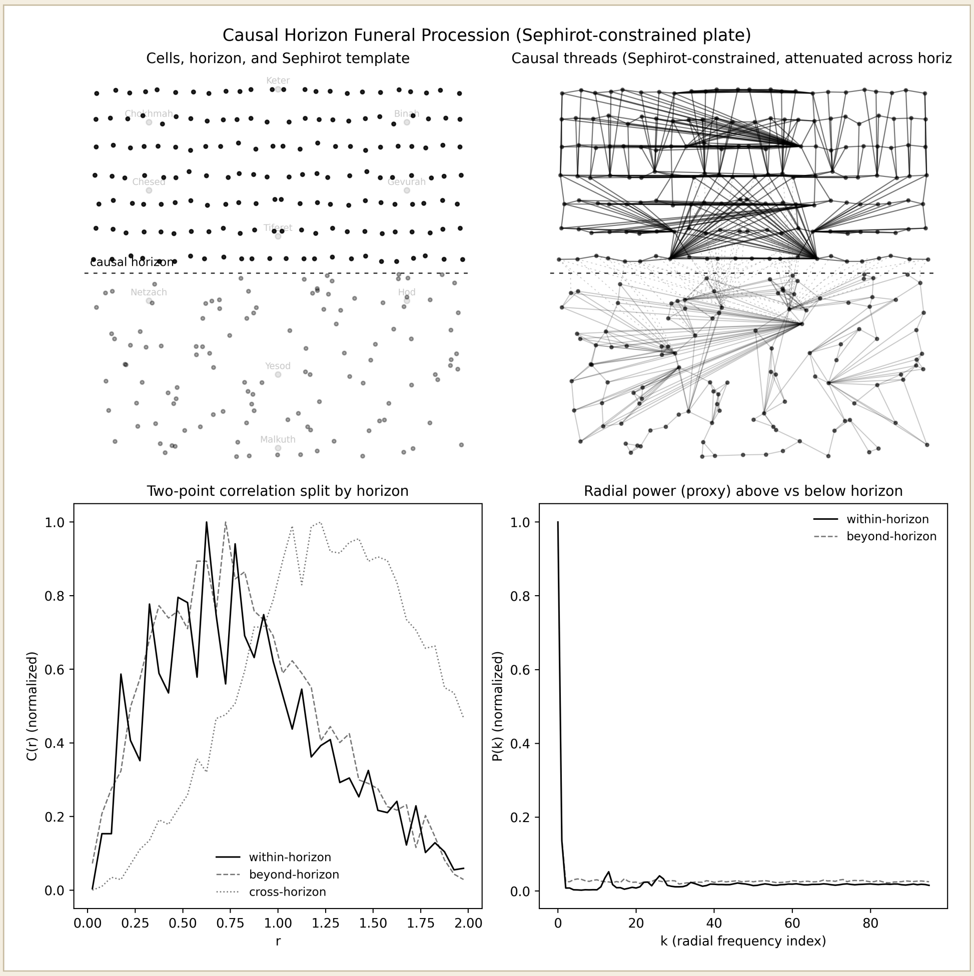


Fig. 5 : Plate v2 — coherent density and increased order under constraints

`p = 0.350  q = 0.350  σ = 0.250  κ = 0.150`

OBSERVED Density becomes more coherent and order increases under constraints.

INTENT Preserve visual causal readability via Sephirot constraints.

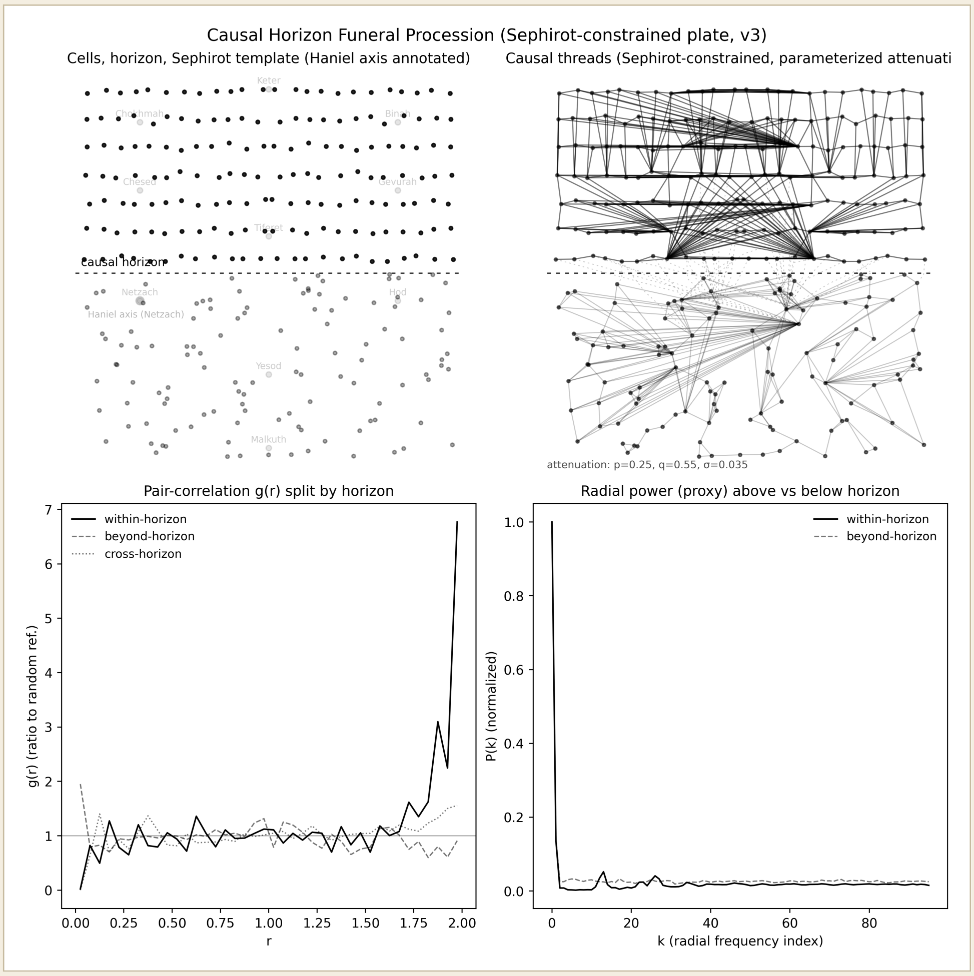


Fig. 6： Plate v3 — locality and haze in balance (p–q shift)

`p = 0.250  q = 0.550  σ = 0.035  κ = 0.150`

OBSERVED A balance emerges between locality and haze.

INTENT Let hardness and ambiguity coexist through p, q, and σ.

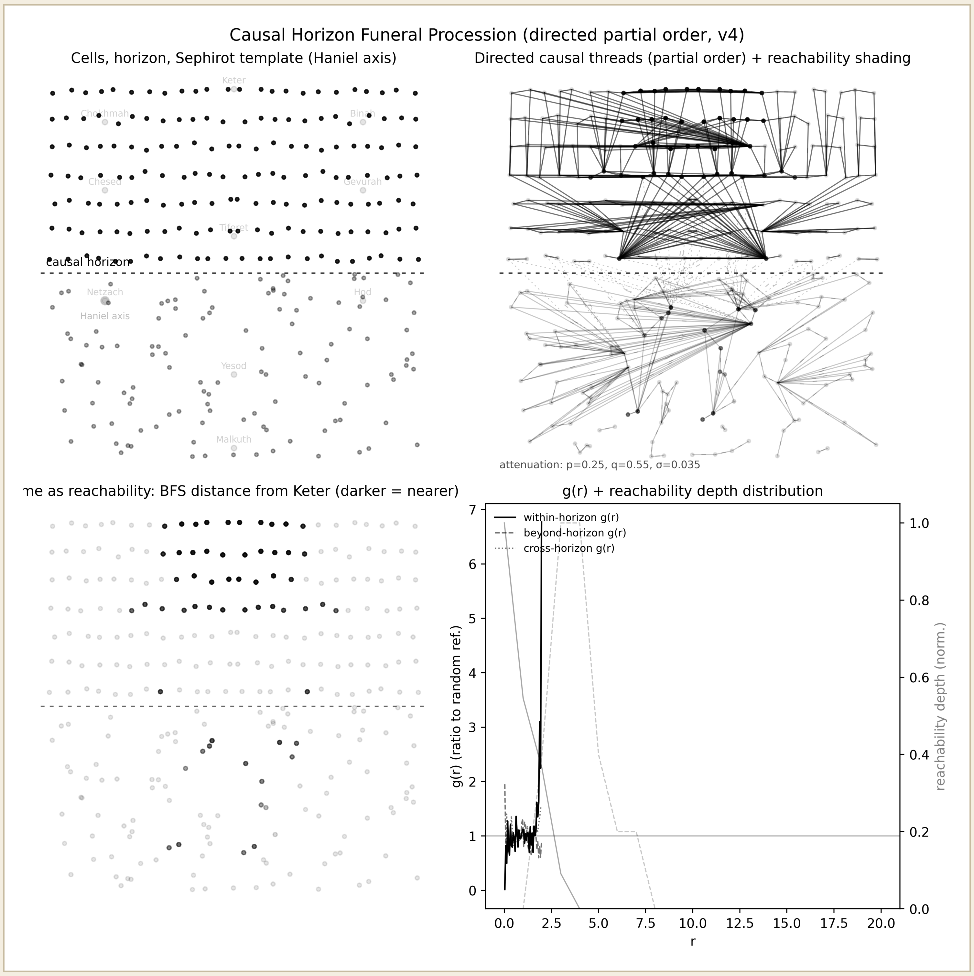


Fig. 7 : Plate v4 — reachability becomes more legible

`p = 0.250  q = 0.550  σ = 0.035  κ = 0.150`

OBSERVED Reachability becomes more legible.

INTENT Reflect causal directionality in the visualization.

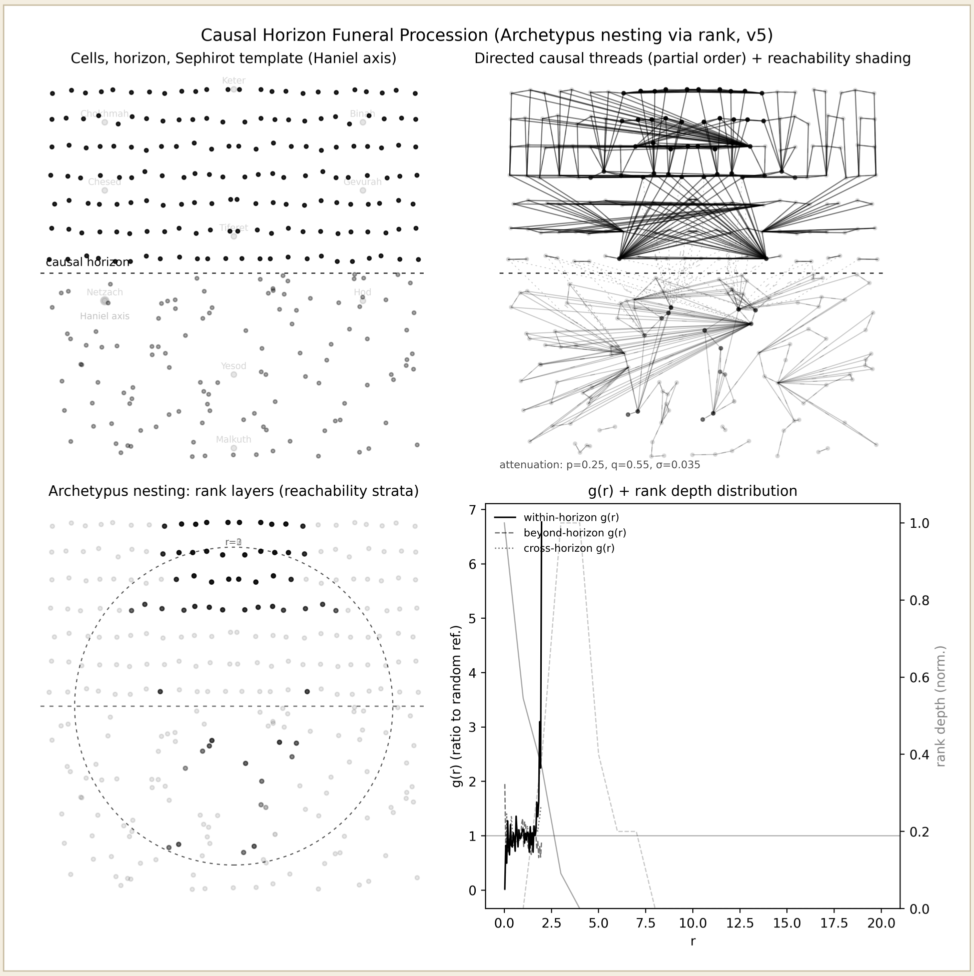


Fig. 8 : Plate v5 — rank repetition becomes prominent

`p = 0.250  q = 0.550  σ = 0.035  κ = 0.150`

OBSERVED Rank repetition becomes prominent.

INTENT Organize the “procession” as a nested, repeating structure.

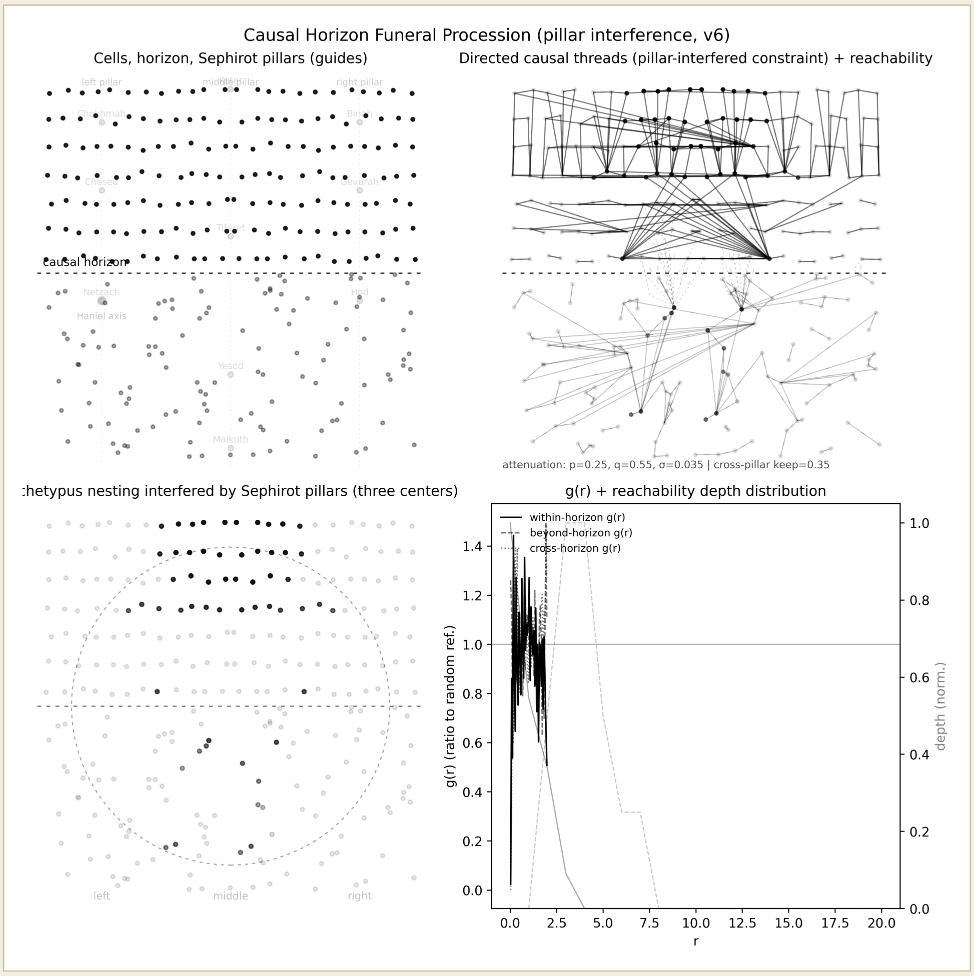


Fig. 9：Plate v6 — κ controls cross-horizon interference

`p = 0.250  q = 0.550  σ = 0.035  κ = 0.350`

OBSERVED Cross-horizon interference becomes controllable.

INTENT Treat “tearing/breaches” as an intentional, κ-mediated effect.

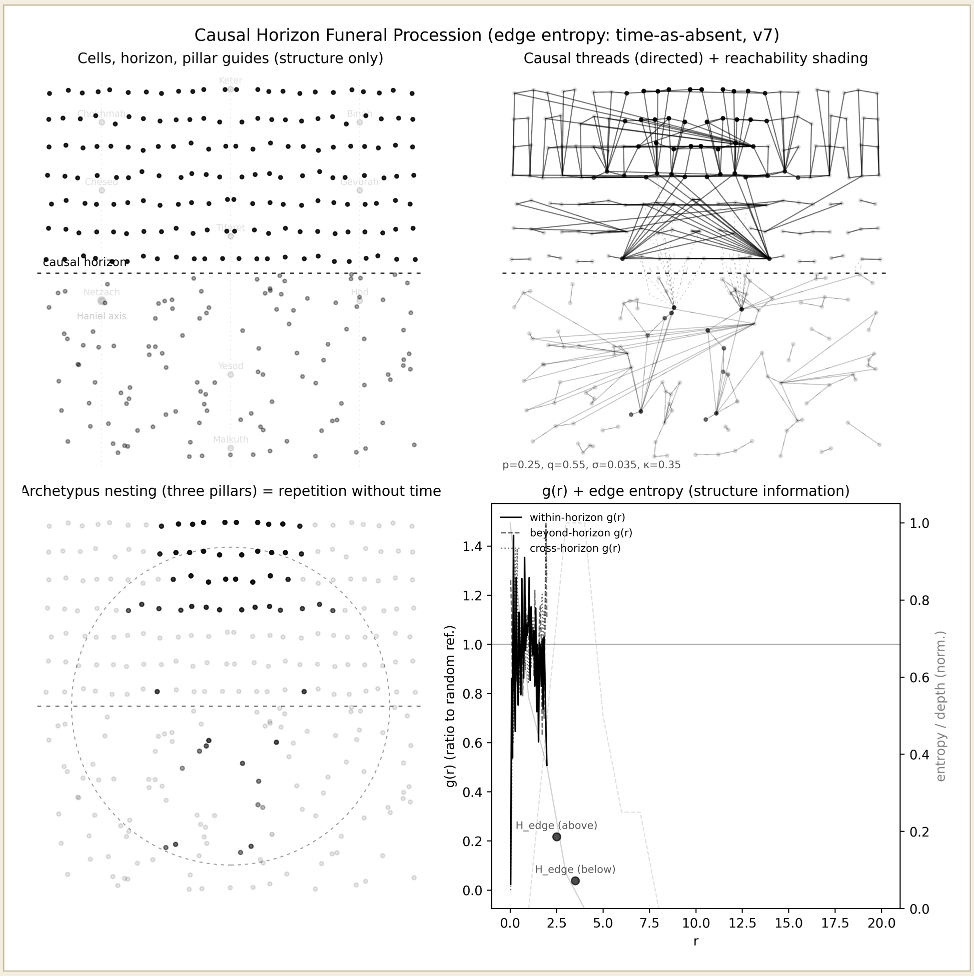


Fig. 10: Plate v7 — diversity maintained via edge entropy

`p = 0.250  q = 0.550  σ = 0.035  κ = 0.350`

OBSERVED Monotony becomes less likely.

INTENT Maintain diversity via edge entropy.

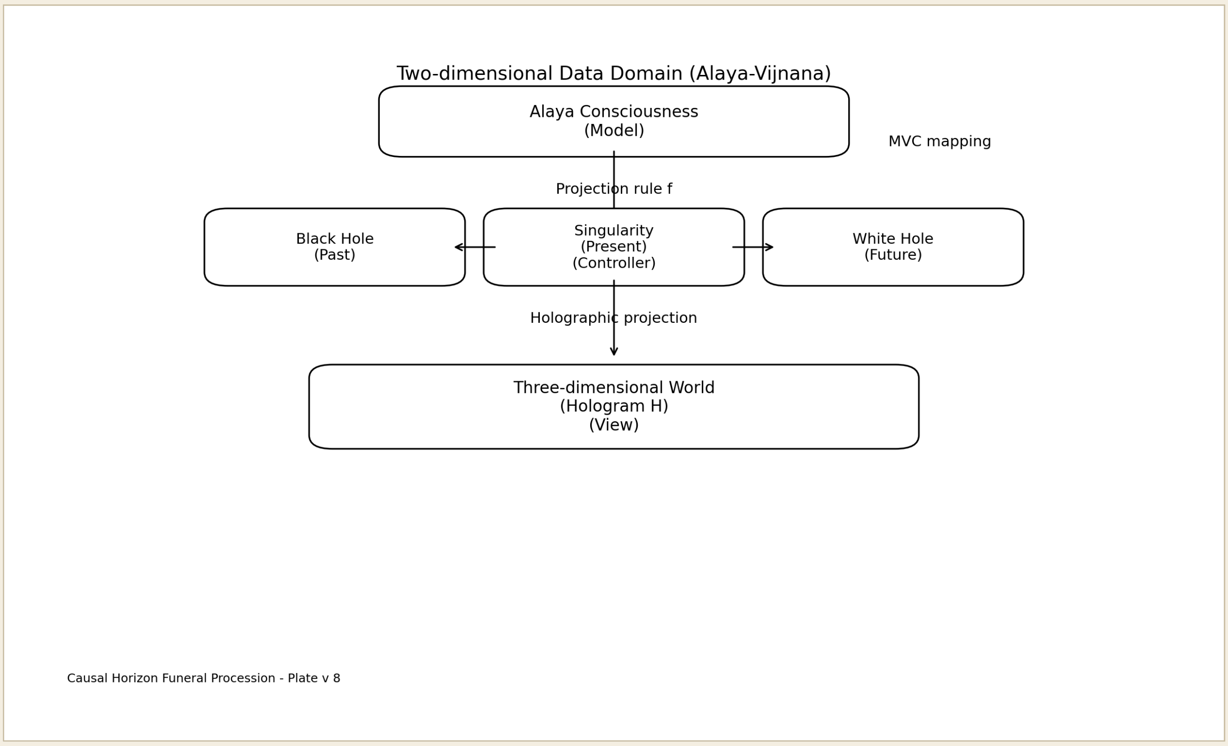


Fig. 11: Plate v8 — Model → Output mapping clarified

`p = 0.500  q = 0.450  σ = 0.350  κ = 0.280`

OBSERVED The Model → Output correspondence becomes clearer.

INTENT Make the projection bridge explicit.

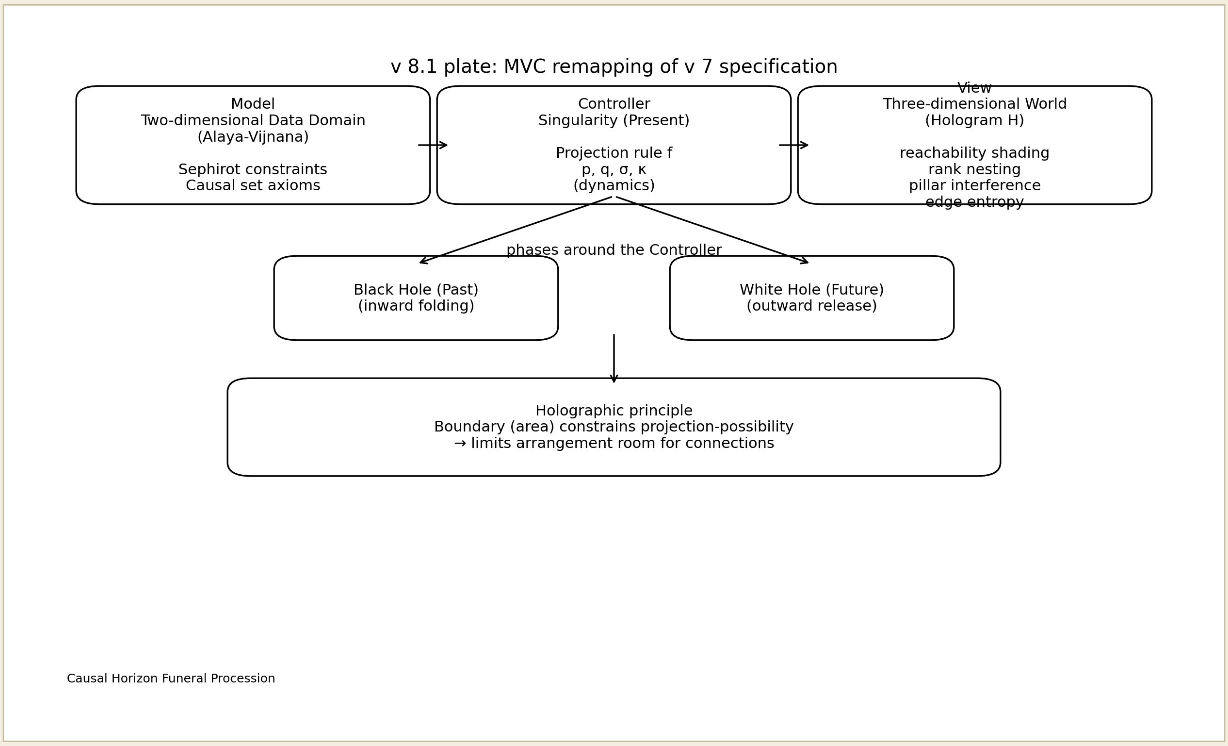


Fig. 12: Plate v8.1 — interpretation stabilizes; MVC correspondence fixed

$p = 0.500 \quad q = 0.450 \quad \sigma = 0.350 \quad \kappa = 0.280$

OBSERVED Interpretation stabilizes.

INTENT Recover v7 and fix the MVC correspondence.

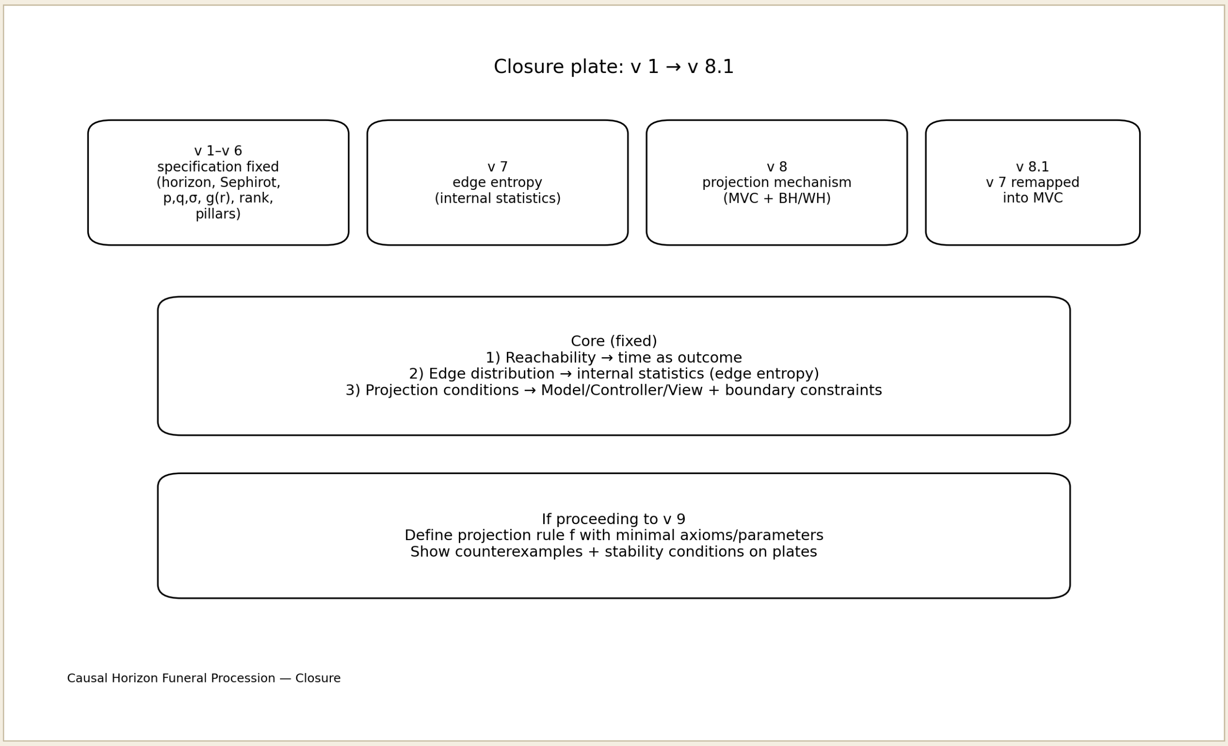


Fig. 13 : Closure — integrated trajectory (v1–v8.1)

$p = 0.500 \quad q = 0.450 \quad \sigma = 0.350 \quad \kappa = 0.280$

OBSERVED The overall trajectory is presented in an integrated view.

INTENT Conclude with a panoramic overview of v1–v8.1.

Closing

This complete edition is an attempt to close the v1–v9 trajectory—rooted in the artwork “Funeral Procession”—under a single axiom system. The four outputs (plate as vision, paper as specification, note as narrative, SNS as sharing) were different surfaces of one structure derived from the same internal state. The claim that the projection rule f is uniquely determined by five axioms is less the completion of a proof than the starting point of further questions: a holographic derivation of κ_{max}, the behavior of f at the boundary, an axiomatization of the Sephirot constraints, and a connection to cultural-evolution theory (A4 = the preservation of diversity). These are left open, for the day a continuation is written.

Appendix A — Reference Code (render_plate)

The complete plate-generation implementation summarized in § 1 is given below. It assumes the same helper functions (knn_edges, pair_correlation, radial_power) and the same internal state (seed and parameter set) as in the main text.

```
def render_plate(filename, seed=42, horizon_y=0.0, dpi=420):
    rng = np.random.default_rng(seed)

    po = perturbed_hex_points(seed=seed)
    po = po[po[:, 1] >= horizon_y]
    pr = random_points(seed=seed + 1)
    pr = pr[pr[:, 1] < horizon_y]

    if len(po) < 140:
        extra = po.copy(); extra[:, 0] *= -1
        po = np.vstack([po, extra]); po = po[po[:, 1] >= horizon_y]

    po = po[:180]; pr = pr[:180]
    pts = np.vstack([po, pr]); n = len(pts)
    edges = knn_edges(pts, k=4)
    above = set(np.where(pts[:, 1] >= horizon_y)[0].tolist())
    below = set(np.where(pts[:, 1] < horizon_y)[0].tolist())

    pair_samples = 20000
    a = rng.integers(0, n, size=pair_samples)
    b = rng.integers(0, n, size=pair_samples)
    m = a != b; a = a[m]; b = b[m]
    dist = np.sqrt(((pts[a] - pts[b]) ** 2).sum(axis=1))

    dA, dB, dC = [], [], []
    for i in range(len(dist)):
        ia, ib = int(a[i]), int(b[i])
        if ia in above and ib in above:   dA.append(dist[i])
        elif ia in below and ib in below: dB.append(dist[i])
        else:                             dC.append(dist[i])

    rA, cA = pair_correlation(np.array(dA), r_max=2.0, bins=40)
    rB, cB = pair_correlation(np.array(dB), r_max=2.0, bins=40)
    rC, cC = pair_correlation(np.array(dC), r_max=2.0, bins=40)
    kA, pA = radial_power(po, grid=192)
    kB, pB = radial_power(pr, grid=192)

    fig = plt.figure(figsize=(10, 10), constrained_layout=True)
    gs = GridSpec(2, 2, figure=fig)

    ax1 = fig.add_subplot(gs[0, 0]); ax1.set_aspect("equal"); ax1.axis("off")
    ax1.scatter(po[:, 0], po[:, 1], s=8, color="black", alpha=0.85)
    ax1.scatter(pr[:, 0], pr[:, 1], s=8, color="black", alpha=0.35)
    ax1.plot([-1.05, 1.05], [horizon_y, horizon_y],
             linestyle=(0, (3, 4)), linewidth=1.0, color="black", alpha=0.8)
    ax1.text(-1.02, horizon_y + 0.04, "causal horizon", fontsize=9)
    ax1.set_xlim(-1.1, 1.1); ax1.set_ylim(-1.1, 1.1)
    ax1.set_title("Cells and causal horizon", fontsize=11)

    ax2 = fig.add_subplot(gs[0, 1]); ax2.set_aspect("equal"); ax2.axis("off")
```

```
ax2.plot([-1.05, 1.05], [horizon_y, horizon_y],
         linestyle=(0, (3, 4)), linewidth=1.0, color="black", alpha=0.8)
for (i, j) in edges:
    xi, yi = pts[i]; xj, yj = pts[j]
    crosses = (yi - horizon_y) * (yj - horizon_y) < 0
    if crosses:
        mx, my = 0.5 * (xi + xj), 0.5 * (yi + yj)
        jx = mx + rng.normal(0, 0.03); jy = my + rng.normal(0, 0.03)
        ax2.plot([xi, jx, xj], [yi, jy, yj], linewidth=0.6,
                 alpha=0.25, color="black", linestyle=(0, (2, 4)))
    else:
        if yi >= horizon_y and yj >= horizon_y:
            ax2.plot([xi, xj], [yi, yj], linewidth=0.8, alpha=0.55, color="black")
        else:
            ax2.plot([xi, xj], [yi, yj], linewidth=0.7, alpha=0.22, color="black")
ax2.scatter(pts[:, 0], pts[:, 1], s=6, color="black", alpha=0.6)
ax2.set_xlim(-1.1, 1.1); ax2.set_ylim(-1.1, 1.1)
ax2.set_title("Causal threads (attenuated across horizon)", fontsize=11)

ax3 = fig.add_subplot(gs[1, 0])
ax3.plot(rA, cA, linewidth=1.2, color="black")
ax3.plot(rB, cB, linewidth=1.0, color="black", alpha=0.55, linestyle="--")
ax3.plot(rC, cC, linewidth=1.0, color="black", alpha=0.55, linestyle=":")
ax3.set_xlabel("r"); ax3.set_ylabel("C(r) (normalized)")
ax3.set_title("Two-point correlation split by horizon", fontsize=11)
ax3.legend(["within-horizon", "beyond-horizon", "cross-horizon"],
           frameon=False, fontsize=9)

ax4 = fig.add_subplot(gs[1, 1])
ax4.plot(kA, pA, linewidth=1.2, color="black")
ax4.plot(kB, pB, linewidth=1.0, color="black", alpha=0.55, linestyle="--")
ax4.set_xlabel("k (radial frequency index)"); ax4.set_ylabel("P(k) (normalized)")
ax4.set_title("Radial power (proxy) above vs below horizon", fontsize=11)
ax4.legend(["within-horizon", "beyond-horizon"], frameon=False, fontsize=9)

fig.suptitle("Causal Horizon Funeral Procession (monochrome plate)", fontsize=13)
fig.savefig(filename, dpi=dpi)
plt.close(fig)
```