Learning Time-Varying Relational Constraints on Dynamic Heterogeneous Graphs

Anonymous authors
Paper under double-blind review

Abstract

I study when changing rules that link related nodes on a typed, time-varying graph can be recovered from the observations, and which kind of model is needed in each case. I call this change relational-semantics drift: two nodes should still satisfy $x_v \approx R^\star_{r,t} x_u$, but the map $R^\star_{r,t}$ can move even when the graph shape and node features look similar. What can be recovered depends on what the data still contains. If each node is reduced to a mixed average over all neighbor types, the product of unlinked node-table marginals no longer depends on the typed maps (Theorem 2(i)). Linked mixed pairs still carry a degree-diluted copy of the typed cross-moment; empirically that SNR is too low for Procrustes (Table 2). The raw typed pairs (x_u, x_v) keep a well-conditioned map: Procrustes recovers it on $\mathrm{SO}(d)$ (Proposition 1), and least squares recovers it on $\mathrm{GL}(d)$ (Proposition 2). A Cayley sheaf with a slow path and a fast path improves anomaly scores on the single-seed smooth benchmark, beating TGN, DualPath, and aggregated Procrustes and approaching EvolveGCN; it tracks coboundary energy, not a typed $\widehat{R}_r$. Averaging nodes first does not recover unlabeled jumps (0.594 ± 0.071 vs. DualPath 0.590 ± 0.062). When the pairs themselves are observed, unlabeled trimmed least squares in $\mathrm{GL}(d)$ recovers the post-jump residual (five-seed AUC 1.000); orthogonal Procrustes fails if the jump leaves $\mathrm{SO}(d)$.

1 Introduction

Graph learning systems that score risk, on transaction networks operate under changing rules. A high-volume transfer between two accounts may be ordinary under one product mix or reporting threshold but anomalous after a policy change or regime change. In such environments the relational semantics of an edge type—what consistency between incident nodes should mean: a conversion rate, a reporting threshold, a product mix—can drift even when the graph topology is largely stable. A useful analogy is a multi-currency ledger whose account graph is fixed. Each instrument is a conversion $x_v \approx R_r x_u$: a USD wire and a EUR wire can look similar in volume while obeying different maps, and a reporting threshold or FX regime can change one conversion while the same accounts still transact. Pooling every counterparty of an account into one mixed vector is like adding dollars to euros before looking up the rate—the unlinked mixed node table is $\mathbf{R}$-invariant (Theorem 2(i)). Edge-linked mixed pairs retain a $1/(\deg(u)\deg(v))$ shadow of the typed cross-moment; that SNR is not enough for Procrustes in Table 2. Only the raw typed pairs (who paid whom, in which instrument) retain a well-conditioned map.

Existing dynamic graph models occupy three nearby cells of this design space and leave a fourth empty. Temporal GNNs such as TGN (Rossi et al., 2020), TGAT (Xu et al., 2020), and DyGFormer (Yu et al., 2023) evolve node memories or temporal attention while keeping relation operators fixed. Parameter-evolving models such as EvolveGCN (Pareja et al., 2020) and ROLAND (You et al., 2022) let weights change over time. Those weights can implicitly absorb smooth operator drift—EvolveGCN reaches 0.932 ± 0.010 post-AUC on my slow-rotation synthetic setting—but they do not isolate relational semantics as a typed geometric object. Heterogeneous GNNs such as R-GCN (Schlichtkrull et al., 2018) and HGT (Hu et al., 2020) encode typed relations, yet are most often trained as static or

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

piecewise-stationary models. None of these families treats the constraint operator itself as the primary time-varying object.

I formalize this gap as relational-semantics drift: a latent typed operator $R^\star_{r,t}$ defines compatibility, but the observation $O(\mathcal{G}_t, x_t)$ may destroy the information needed to recover it. What can be recovered therefore depends on what survives observation, not only on which GNN is trained. Building on cellular sheaves on graphs (Hansen & Gebhart, 2020; Bodnar et al., 2022), I assign stalks to nodes and a Cayley-parameterized connection on $\mathrm{SO}(d)$ to edges. The sheaf coboundary δ_t measures disagreement after restriction; its squared norm is used as a first-class detection signal on Lipschitz pieces. Smooth evolution is a GRU on the Lie algebra; a snapshot encoder proposes a Cayley update after a detected change; the two are fused by Cayley-chart interpolation. That architecture improves anomaly prediction under a planted smooth relational process; it does not emit a typed $\widehat{R}_r$. Unlabeled jump recovery is a different problem: it requires endpoint pairs and a model class that matches the planted relative map, which need not remain in $\mathrm{SO}(d)$.

Contributions. (i) I prove that the product of mixed-type node-table marginals cannot identify typed operators (Theorem 2(i)); edge-linked mixed pairs are not claimed invariant. (ii) I show that raw typed endpoint pairs do identify the operator: $\mathrm{SO}(d)$ via Procrustes (Proposition 1), $\mathrm{GL}(d)$ via OLS (Proposition 2), with a standard finite-sample rate (Lemma 1). (iii) On one generator seed, a Cayley-parameterized sheaf improves smooth-drift anomaly prediction relative to TGN, DualPath, and aggregated Procrustes and approaches EvolveGCN; multi-seed jump results are reported separately. (iv) I give an informal energy error budget (Theorem 1) on Lipschitz pieces and validate the robust pair estimator IT-GLLS empirically (AUC 1.000±0.000 across five seeds). (v) On Elliptic++ I report a robustness–accuracy tradeoff, not a causal identification of $R^\star_t$.

2 Related Work

Temporal and dynamic GNNs. Rossi et al. (2020) maintain per-node memory with time encoding; Xu et al. (2020) use temporal attention; Yu et al. (2023) apply transformer-style encoders to interaction sequences. Pareja et al. (2020) evolve GCN weights with an RNN; You et al. (2022) track live representations across snapshots. Surveys of dynamic graph learning emphasize evolving structure and features (Kazemi et al., 2020; Skarding et al., 2021). These methods adapt states or parameters, not typed relational constraints.

Heterogeneous GNNs. Schlichtkrull et al. (2018) use a separate weight matrix per relation; Hu et al. (2020) use type-dependent attention. Relation types are typically treated as fixed categories: the type label does not change meaning over time.

Sheaf neural networks. Sheaf networks replace the graph Laplacian with a sheaf Laplacian defined by restriction maps, improving behaviour under heterophily and providing a geometric notion of local consistency (Hansen & Ghrist, 2019; Hansen & Gebhart, 2020; Bodnar et al., 2022; Barbero et al., 2022). Concurrent temporal sheaf work equips each node with a time-varying orthogonal frame and transports states for causal link prediction (Asif et al., 2026). That addresses coordinate alignment of memories. I instead treat a shared Cayley connection as an energy tracker on Lipschitz pieces, and typed endpoint pairs as the observation that identifies relation-specific maps.

Continuous dynamics and jumps. Neural ODEs and latent ODEs model smooth latent trajectories (Chen et al., 2018; Rubanova et al., 2019); neural CDEs handle irregular series (Kidger et al., 2020); neural jump ODEs add discontinuities (Herrera et al., 2021). My Cayley path is discrete-time (GRU + snapshot encoder) on Lipschitz pieces; unlabeled discontinuities are handled by a pair-level $\mathrm{GL}(d)$ residual when endpoint features are observed (Section 4.4).

Drift, continual learning, and AML graphs. Concept-drift detectors such as ADWIN and the broader drift literature monitor $P(X)$ or $P(Y \mid X)$ (Bifet & Gavaldà, 2007;

Gama et al., 2014). Continual learning seeks to retain predictive performance under non-stationarity (Parisi et al., 2019). AML graph methods include Elliptic/Elliptic++ analyses, AMLSim, PaySim, and LaundroGraph (Weber et al., 2019; Elmisaad et al., 2023; Weber et al., 2018; Lopez-Rojas et al., 2016; Cardoso et al., 2022). I do not claim a new AML typology detector. Whether a given real dataset exhibits relational-semantics drift is an empirical question (Section 6), not a corollary of a temporal AUC drop.

3 Problem Formulation

3.1 Dynamic heterogeneous graphs

At time t I observe a heterogeneous graph

$$\mathcal{G}_t = (V_t, E_t, \tau_v, \tau_e, x_t), \tag{1}$$

where $E_t \subseteq V_t \times V_t$ is a set of directed ordered pairs (u, v), written $u \to v$, $\tau_v \colon V_t \to \mathcal{T}_V$ and $\tau_e \colon E_t \to \mathcal{T}_E$ are node- and edge-type maps, and $x_t(v) \in \mathbb{R}^{d_x}$ is a node feature. The coboundary restriction maps below are assigned with respect to this fixed source-to-target orientation. Edges may carry timestamps and amounts; I write $a_t(e)$ for optional edge attributes. Supervision consists of sparse binary labels $y_t(e)$ or $y_t(v)$ (e.g., illicit transactions or SAR indicators) on a subset of entities and windows.

3.2 Cellular sheaf and coboundary

A cellular sheaf $\mathcal{F}_t$ on $\mathcal{G}_t$ assigns a stalk $\mathcal{F}_t(v) \cong \mathbb{R}^d$ to each node and an edge stalk $\mathcal{F}_t(e) \cong \mathbb{R}^d$. Restriction maps send node stalks to the edge stalk. I do not use a shared left action $\rho s_u - \rho s_v$.

Definition 1 (Coboundary violation). For a 0-cochain s_t with $s_t(v)$ in the stalk of v (an encoding of x_t),

$$(\delta_t s_t)(u \to v) \;=\; \rho_t\, s_t(u) \;-\; \rho_t^{\top}\, s_t(v), \tag{2}$$

where $\rho_t \in \mathrm{SO}(d)$ is a learned shared relative transport. The energy $\|(\delta_t s_t)(e)\|^2$ is the constraint violation on edge e.

The typed formulation ($\rho_{r,t}$ per relation type) is used in Theorem 2 to characterize the observation conditions required for identifying relation-specific operators; the neural experiment evaluates shared-operator energy tracking rather than relation-specific operator recovery. Extension to per-type $\rho_{r,t}$ is discussed in Section 7.

This is an asymmetric discrete connection (Barbero et al., 2022): source and target use ρ and $\rho^{\top}$, not the same left action. If both maps were ρ, orthogonality would cancel the operator ($\|\rho s_u - \rho s_v\| = \|s_u - s_v\|$) and ρ could not affect the anomaly energy. Because the two restriction maps differ, reversing the assigned orientation of $u \to v$ does not merely negate the coboundary: for a general $\rho \in \mathrm{SO}(d)$, $\left\|\rho s_v - \rho^{\top} s_u\right\| \neq \left\|\rho s_u - \rho^{\top} s_v\right\|$. The coboundary energy is therefore sensitive to the directedness of E_t. The relative map of the $(\rho, \rho^{\top})$ connection is $R_t = \rho_t^2$, not ρ_t itself. Throughout this paper, when I refer to the learned relational operator of the Cayley path, I mean the induced relative transport $R_t = \rho_t^2$. The coboundary energy $\left\|\rho s_u - \rho^{\top} s_v\right\|^2$ vanishes on compliant pairs and is large on anomalous ones, making it a valid anomaly statistic, but ρ_t should not be compared directly to $R_t^{\star}$ without accounting for the square-root relationship. Because $\rho \in \mathrm{SO}(d)$, left-multiplication by ρ is an orthogonal action and $\left\|\rho s_u - \rho^{\top} s_v\right\| = \|R_t s_u - s_v\|$. Equation equation 2 is therefore equivalent to comparing the transported source $R_t s_u$ with the target s_v, matching the generative rule $x_v \approx R_t^{\star} x_u$. Equivalently, left-multiplication by $\rho^{\top}$ compares s_u with $R_t^{\top} s_v$, not with $R_t s_v$. The generator plants a pair $(\rho^{\mathrm{src}}, \rho^{\mathrm{tgt}})$ and labels from $\|\rho^{\mathrm{src}} x_u - \rho^{\mathrm{tgt}} x_v\|^2$. Compliant pairs therefore obey $x_v \approx R_t^{\star} x_u$ with $R_t^{\star} = (\rho^{\mathrm{tgt}})^{+} \rho^{\mathrm{src}}$. Smooth Givens evolution keeps $R_t^{\star}$ near $\mathrm{SO}(d)$. Jumps are additive ($\rho \leftarrow \rho + \Delta$ with Δ skew), so $R_t^{\star}$ can leave $\mathrm{SO}(d)$. The Cayley model compresses the pair into one $\mathrm{SO}(d)$ connection; Section 4.4 scores the $\mathrm{GL}(d)$ residual when pairs are observed.

3.3 Four drift types

I distinguish four sources of non-stationarity, nested in the usual concept-drift taxonomy (Gama et al., 2014). Feature drift is a change in the node-feature law $P(x_t)$; structure drift is a change in $P(E_t)$; marginal label drift (prior shift) is a change in $P(y_t)$. Concept drift is a change in the posterior $P(y_t \mid \mathcal{G}_t, x_t)$. Under Assumption 1, labels are a function of coboundary energy, so evolution of the latent relative maps $R^\star_{r,t}$ induces a structured posterior shift: I call this subclass relational-semantics drift. It is distinct from unconstrained label-policy shifts that move $P(y_t \mid \mathcal{G}_t, x_t)$ for reasons other than those maps. Under orthogonal typed drift, mixed-type node marginals can remain stationary (Theorem 2) while the joint pairwise law of typed endpoints (x_u, x_v) still changes. The last is the object of study: two graphs with similar degrees and feature margins can induce different risk if $R^\star_{r,t}$ has jumped.

Definition 2 (Relational-semantics drift learning). Given $\{(\mathcal{G}_t, x_t, y_t)\}_{t\in[0,T]}$, learn a family of typed relational maps $\{R_{r,t}\}_{r\in\mathcal{T}_E,\, t\in[0,T]}$ such that (i) each $t \mapsto R_{r,t}$ is piecewise smooth with finitely many jumps, and (ii) a residual of the pair constraint $x_v \approx R_{r,t}x_u$ (coboundary energy, Procrustes residual, or $\mathrm{GL}(d)$ residual) is predictive of y_t. The neural Cayley path uses a single shared connection $\rho_t \in \mathrm{SO}(d)$ with induced transport $R_t = \rho_t^2$ as an operational simplification for shared coboundary-energy tracking; Theorem 2 and IT-GLLS use the typed family $\{R_{r,t}\}$.

I do not treat $\|\delta_t s_t\|$ merely as a regularizer on embeddings. It is the detection statistic.

4 Method

4.1 Orthogonal restriction maps

Unconstrained maps can collapse or become ill-conditioned, making coboundary norms uninformative. The submitted neural model uses a shared ρ_t across edge types. I constrain that connection to $\mathrm{SO}(d)$ using the Cayley transform (Gallier, 2011). For a skew-symmetric matrix $A \in \mathfrak{so}(d)$,

$$\mathrm{Cayley}(A) \;=\; (I - A)(I + A)^{-1} \in \mathrm{SO}(d), \tag{3}$$

which is defined for every real skew A (since $v^\top(I+A)v = \|v\|^2 > 0$) and is a diffeomorphism onto $\mathrm{SO}(d)$ minus $\{R : -1 \in \mathrm{spec}(R)\}$. The inverse is $\mathrm{Cayley}^{-1}(R) = (I - R)(I + R)^{-1}$ on that image. All ρ updates are composed on the group, so $\rho_t \in \mathrm{SO}(d)$ by construction.

4.2 Slow path: recurrent Lie-algebra evolution

A GRU consumes a summary of the current coboundary field and hidden state h_{t-1} and emits Lie-algebra coefficients. These are assembled into a skew matrix $A_t \in \mathfrak{so}(d)$ and mapped to a relative increment $\Delta\rho_t = \mathrm{Cayley}(A_t)$ from the identity (not an absolute target on $\mathrm{SO}(d)$). A residual gate $g_t^{\mathrm{slow}} \in [0.1, 1]$ interpolates that increment in the Cayley chart:

$$\rho_t^{\mathrm{slow}} \;=\; \rho_{t-1}\,\mathrm{Cayley}\big(g_t^{\mathrm{slow}} \cdot \mathrm{Cayley}^{-1}(\Delta\rho_t)\big) \;=\; \rho_{t-1}\,\mathrm{Cayley}(g_t^{\mathrm{slow}} A_t). \tag{4}$$

When $g_t^{\mathrm{slow}} = 1$ this is left composition $\rho_{t-1}\Delta\rho_t$, so history accumulates. This path has long memory and is well suited to Lipschitz drift, but it is vulnerable to memory poisoning: after a jump, h_t still encodes the previous regime.

4.3 Fast path: snapshot encoder

A two-layer message-passing encoder maps $(\mathcal{G}_t, x_t)$ to a graph embedding and then to a shared Lie-algebra increment (copied across types), yielding $\rho_t^{\mathrm{fast}} = \mathrm{Cayley}(A_t^{\mathrm{fast}})$. This path has no recurrent state: it cannot be poisoned by pre-jump history, but a Cayley snapshot cannot represent a relative map that has left $\mathrm{SO}(d)$.

4.4 Unlabeled GL(d_x) snapshot residual

When raw endpoint pairs $(x_u, x_v) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_x}$ are observed, I fit one $C_{r,t} \in \mathrm{GL}(d_x)$ per type by iteratively trimmed least squares (IT-GLLS): ordinary least squares on all pairs, drop the top α=0.15 residuals, refit (three iterations). No labels and no jump indicator are used. The residual $r_e = \|x_v - C_{r,t}x_u\|$ is concatenated to the edge head. This is not Cayley fusion: $C_{r,t}$ is unconstrained in ambient feature space, distinct from the stalk connection $\rho_t \in \mathrm{SO}(d)$. Anomalous targets in the generator are independent of sources, so unlabeled OLS is already nearly consistent; trimming stabilizes the worst seed.

4.5 Change detection and Cayley interpolation

The detector scores how badly the new snapshot violates yesterday's maps, $c_t^{\mathrm{old}} = \|\delta[\rho_{t-1}]\, s_t\|$. Let $\bar{c}_t$ be an EMA of c_t^{old} and $\Delta_t = c_t^{\mathrm{old}} - \bar{c}_t$. A jerk pulse is raised when $|\Delta_t - \Delta_{t-1}|/(|\bar{c}_t| + \varepsilon) > \tau_{\mathrm{jerk}}$; then $g_t^{\mathrm{rec}} = \max(0, 1 - k/W)$ for W steps. The snapshot embedding z_t drives

$$d_t = 1 - \cos(z_t, \bar{z}_{t-1}), \quad g_t^{\mathrm{emb}} = \sigma(w d_t + b), \quad \bar{z}_t = \beta \bar{z}_{t-1} + (1-\beta) z_t, \tag{5}$$

and the unsupervised fusion weight is $g_t^{\mathrm{fuse}} = \max(g_t^{\mathrm{emb}}, g_t^{\mathrm{rec}})$ (jerk enters only through g_t^{rec}; defaults in Appendix C). The oracle diagnostic forces $g_t^{\mathrm{fuse}} = 1$ on planted jump labels. The fused map interpolates the two absolute connections in the Cayley chart:

$$\rho_t \;=\; \rho_t^{\mathrm{slow}}\, \mathrm{Cayley}\big(g_t^{\mathrm{fuse}} \cdot \mathrm{Cayley}^{-1}\big((\rho_t^{\mathrm{slow}})^\top \rho_t^{\mathrm{fast}}\big)\big). \tag{6}$$

Ambient convex combinations leave $\mathrm{SO}(d)$; this interpolant does not. Shrinking the fast Lie algebra by 0.1 keeps ρ_t^{fast} near I, not the relative rotation $(\rho_t^{\mathrm{slow}})^\top \rho_t^{\mathrm{fast}}$, and limits large snapshot jumps (Appendix C). Hard-handoff—copy ρ_t^{fast} into the diffusion connection and the persistent slow state before scoring, and reset the GRU—is oracle-only.

4.6 Sheaf diffusion and scoring

A learned $P{:}\,\mathbb{R}^{d_x} \to \mathbb{R}^d$ yields stalks $s_t = Px_t$. Three degree-normalized restriction-map layers with skip connections produce endpoint embeddings (Appendix C); this is not $(I - \eta\delta^\top\delta)XW$. An MLP $(2d[+1]) \to 64 \to 32 \to 1$ scores edges and, when pairs are available, consumes r_e.

4.7 Objectives and routing curriculum

$$\mathcal{L} \;=\; \mathcal{L}_{\mathrm{BCE}} + \lambda_{\mathrm{s}}\mathcal{L}_{\mathrm{smooth}} + \lambda_{\mathrm{j}}\mathcal{L}_{\mathrm{jump}} + \lambda_{\mathrm{f}}\mathcal{L}_{\mathrm{fast}}. \tag{7}$$

$\mathcal{L}_{\mathrm{BCE}}$ is positive-weighted edge BCE. $\mathcal{L}_{\mathrm{smooth}} = \|\rho_t - \rho_{t-1}\|_{\mathrm{F}}^2$; $\mathcal{L}_{\mathrm{fast}}$ is the mean coboundary norm under ρ_t^{fast}; $\mathcal{L}_{\mathrm{jump}}$ is BCE(g_t^{fuse}, $\mathbf{1}_{\mathrm{jump}}$). Defaults: $\lambda_{\mathrm{s}} = 0.005$, $\lambda_{\mathrm{f}} = 0.1$, $\lambda_{\mathrm{j}} = 0$ (headline) or 1 (oracle), plus $\lambda_o = 0.01$ on $\big\|\rho_t^\top \rho_t - I\big\|_{\mathrm{F}}^2$ (numerically near zero under Cayley). A three-phase curriculum trains the fast path with $g^{\mathrm{fuse}} = 1$, optionally the detector, then fine-tunes (Appendix C). The unsupervised protocol skips jump BCE and hard-handoff.

5 Energy prediction error

Coboundary energies below are evaluated on stalk cochains $s_t(v) \in \mathbb{R}^d$ (in the implemented model, $s_t = Px_t$). Pair identifiability uses raw (or projected) endpoint features in $\mathbb{R}^{d_x}$. A single expected energy $\mathbb{E}\,\|(\delta[\rho]s)(e)\|^2$ is a scalar function on a $\dim \mathrm{SO}(d) = d(d-1)/2$ manifold and cannot identify ρ on an open set. I therefore track the energy that enters the detector, not $\|\widehat{\rho} - \rho^\star\|_{\mathrm{F}}$. Operator recovery is a separate corollary that needs a matrix-valued design: orthogonal Procrustes if $R^\star \in \mathrm{SO}(d)$ (Proposition 1), unconstrained least squares otherwise (Proposition 2). The budget below is silent at jump times and does not justify the GRU.

Assumption 1 (Violation–label coupling). There exists a strictly increasing g that is locally Lipschitz with a locally Lipschitz inverse on a set of positive measure, such that

$$\mathbb{P}(y_t(e) = 1 \mid \mathcal{G}_t, s_t, \rho_t^\star) \;=\; g\big(\,\|(\delta_t^\star s_t)(e)\|^2\,\big). \tag{8}$$

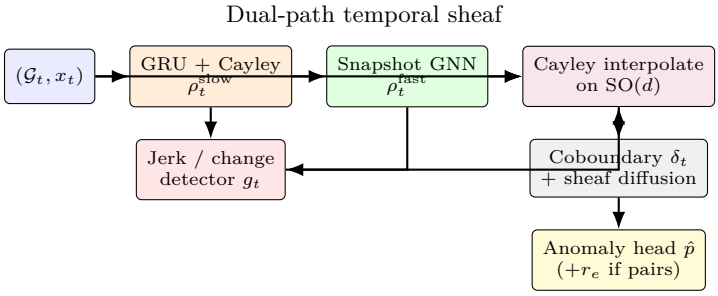

Figure 1: Dual-path architecture. Cayley fusion supplies coboundary energy under Lipschitz SO(d) drift. Unlabeled jumps are scored by a GL(d_x) pair residual r_e when endpoints are observed.

Assumption 2 (Piecewise Lipschitz trajectory). $\rho_t^\star$ is L_{drift}-Lipschitz in Frobenius norm except at at most K jump times.

Assumption 3 (Windowed observations). Each Lipschitz window of length Δ_t contains n_t labeled edges with bounded coboundary energies. Write $\varepsilon_{\text{label}}$ for the deviation between the empirical positive rate and $\frac{1}{n_t}\sum_e g(\|(\delta_t^\star s_t)(e)\|^2)$ (finite-sample label error and label noise).

Write $E_t(\rho) = \frac{1}{n_t}\sum_e \|(\delta[\rho]\, s_t)(e)\|^2$ and $E_t^\star = E_t(\rho_t^\star)$. Let $\widehat{\rho}_t$ denote the connection used by the detector on that window (the implemented Cayley path; not assumed to be an ERM).

Theorem 1 (Energy error budget away from jumps, informal). Under Assumptions 1–3, for t in the interior of a Lipschitz piece the energy gap admits the decomposition

$$\left|E_t(\widehat{\rho}_t) - E_t^\star\right| \;\lesssim\; C_1 n_t^{-1/2} + C_2 \Delta_t L_{\text{drift}} + C_3 \varepsilon_{\text{model}} + C_4 \varepsilon_{\text{label}}. \tag{9}$$

This is a structural budget, not a uniform high-probability bound: the C_1 term is pointwise in ρ, $\varepsilon_{\text{model}}$ excludes optimizer and gate error, and a Jensen remainder from nonlinear g is folded into $\varepsilon_{\text{label}}$ (Appendix A). The budget does not apply at jumps and does not identify $\widehat{\rho}_t$ as a matrix.

Proposition 1 (Procrustes identifiability). If endpoint features are observed as pairs (x_u, x_v) with $x_v = R^\star x_u + \xi$, $R^\star \in \text{SO}(d)$, $\mathbb{E}[x_u x_u^\top]$ full rank, and $\mathbb{E}[\xi x_u^\top] = 0$, the population criterion $\mathbb{E}\,\|Rx_u - x_v\|^2$ has a unique minimizer on SO(d) given by the polar factor of $\mathbb{E}[x_v x_u^\top]$ (orthogonal Procrustes). The implemented coboundary equation 2 is the same pair criterion after replacing R by the connection $(\rho, \rho^\top)$, whose induced relative transport is $R_t = \rho_t^2$. This condition is about the matrix $\mathbb{E}[x_v x_u^\top]$, not a scalar energy. The polar estimator is misspecified if $R^\star \notin \text{SO}(d)$. Theorem 2(i) shows that mixed-type node-table marginals are $\mathbf{R}$-invariant; the identifying matrix is the pooled typed cross-moment of raw pairs.

Proposition 2 (GL identifiability). If $x_v = R^\star x_u + \xi$ with $R^\star \in \text{GL}(d)$, $\mathbb{E}[x_u x_u^\top]$ full rank, and $\mathbb{E}[\xi x_u^\top] = 0$, unconstrained least squares identifies $R^\star$. The polar factor of $\mathbb{E}[x_v x_u^\top]$ is then the wrong estimator.

Lemma 1 (OLS finite-sample rate). Let n compliant pairs satisfy $x_v = R^\star x_u + \xi$ with x_u and ξ independent and sub-Gaussian, ξ with proxy variance σ^2, and let $\Sigma = \mathbb{E}[x_u x_u^\top]$ with $\lambda_{\min}(\Sigma) > 0$. Then the ordinary least-squares estimator satisfies

$$\left\|\widehat{R}_{\text{OLS}} - R^\star\right\|_{\text{F}} \;=\; O_{\mathbb{P}}\big(\sigma\, \lambda_{\min}(\Sigma)^{-1/2} \sqrt{d^2/n}\big). \tag{10}$$

This is the Frobenius rate for a $d \times d$ map: each of the d output coordinates is a d-dimensional OLS problem (Wainwright, 2019), after Gram concentration of $\widehat{\Sigma}$ at $O_{\mathbb{P}}(\sqrt{d/n})$ for sub-Gaussian x_u (Vershynin, 2018, Thm. 4.7.1).

Under Assumption 4 the generator's sources are isotropic Gaussian, so the lemma applies on compliant pairs.

IT-GLLS extends this baseline by iteratively trimming high-residual pairs (α=0.15, three iterations; Section 4.4) to handle unlabeled contamination. A formal breakdown-point analysis would require additional margin conditions on the compliant versus anomalous residual distributions, which I do not assume. The empirical result (AUC 1.000±0.000 across five seeds; Table 5) shows that the estimator is practically effective under the planted DGP.

5.1 Observation design

Assumption 4 (Isotropic pair model). Sources x_u^e are isotropic in $\mathbb{R}^{d_x}$ (law invariant under $\mathrm{SO}(d_x)$). For a compliant edge of type r, $x_v^e = R_r x_u^e + \xi^e$ with $R_r \in \mathrm{SO}(d_x)$ and ξ^e isotropic, independent of x_u^e and of other edges. Anomalous targets, when present, are isotropic and independent of the source. Distinct edges have independent sources (the generator draws a fresh source per edge).

Write $I(w)$ for the multiset of raw endpoint features incident to node w (the source feature of e if w is its source, the target feature if w is its target), and

$$\bar{x}_w \;=\; \frac{1}{|I(w)|} \sum_{z \in I(w)} z \tag{11}$$

for the mixed-type mean aggregate (no type index). This is the node table used by TGN, EvolveGCN, DualPath, and node-aggregated Procrustes in Section 6.

Theorem 2 (Aggregation non-identifiability). Assume Assumption 4 and $R \geq 2$ edge types. Let $\mathbf{R} = (R_1, \ldots, R_R) \in \mathrm{SO}(d_x)^R$.

(i) For every node w, the law of $\bar{x}_w$ does not depend on $\mathbf{R}$. Hence the map $\mathbf{R} \mapsto \prod_w \mathbb{P}_{\mathbf{R}}(\bar{x}_w)$ is constant on $\mathrm{SO}(d_x)^R$ and is not injective.

(ii) The pooled typed cross-moment of raw compliant pairs $S_r = \mathbb{E}[x_v^e (x_u^e)^\top \mid \mathrm{type}(e) = r,\ e \text{ compliant}]$ equals $R_r \Sigma_r$ with $\Sigma_r = \mathbb{E}[x_u^e (x_u^e)^\top \mid r]$. If Σ_r is full rank, S_r identifies R_r by the polar factor of Proposition 1. Independent anomalous targets of rate α scale the unconditional typed moment to $(1-\alpha) R_r \Sigma_r$ and do not change the polar factor. A single-node outer product $x_v x_u^\top$ ($|N_r(u)| = 1$) has rank 1. For $d_x \geq 3$, $\dim \mathrm{SO}(d_x - 1) > 0$, so a rank-1 matrix does not identify R_r on an open set. For $d_x = 2$ the stabilizer is discrete and one nonzero pair determines the rotation angle.

Part (i) is population non-identifiability for estimators that use only the product of unlinked node marginals. Edge-linked mixed pairs $\{(\bar{x}_u, \bar{x}_v)\}$ retain a degree-diluted copy $\mathbb{E}[\bar{x}_v \bar{x}_u^\top \mid e \in E_r] \propto R_r \Sigma_r$; that is a finite-sample SNR issue, not a corollary of (i). Per-relation aggregation (e.g. R-GCN typed messages) is a different statistic and is not covered. The non-identifiability in (i) uses Assumption 4 in an essential way: independent per-edge sources and isotropy make each summand of $\bar{x}_w$ $\mathbf{R}$-invariant in law.

Remark 1 (Empirical counterpart). Table 2 is a finite-sample SNR check after mixed aggregation, not a corollary of Theorem 2(i). Table 4 is the operator-error view on raw pairs. I do not claim that every GNN is non-identifiable, nor that mixed labeled pairs $\{(\bar{x}_u, \bar{x}_v, r_e)\}$ are invariant.

Remark 2 (Persistent node features). On a realistic graph a node w carries a single feature vector reused on every incident edge. Then the summands of equation 11 are dependent, and typed maps can leak into node-level joints, cross-edge covariances, and motifs (e.g. $x_{v_1} = R_1 x_u$ and $x_{v_2} = R_2 x_u$ sharing a source). Theorem 2(i) does not cover that DGP: it is a statement about the product of mixed-type marginals under fresh isotropic endpoint draws, not about persistent entities.

Remark 3 (Jumps, not proved). Theorem 1 does not cover discontinuities, optimization error, or gating mistakes, and does not justify the GRU. Figure 3 is an empirical check, not a corollary of equation 9.

Table 1: Leak-free post-drift edge AUROC on clean planted-constraint graphs, five generator seeds 42–46 (mean±std, Bessel). Train $t < 0.5$; no jump indicators. Mine is the aggregated-node Cayley sheaf without pair residuals. Per-seed jump scores are Table 5.

Drift	MLP	TGN	EvoGCN	Static	DualPath	Mine
Slow smooth	0.831±0.016	0.841±0.013	**0.932±0.010**	0.785±0.038	0.878±0.010	0.930±0.013
Fast smooth	0.780±0.014	0.819±0.010	**0.861±0.013**	0.733±0.031	0.822±0.004	0.828±0.010
Single jump	0.584±0.059	0.583±0.054	0.558±0.036	0.567±0.041	0.590±0.062	**0.594±0.071**
Multi-jump	0.514±0.016	**0.545±0.023**	0.515±0.017	0.531±0.012	0.531±0.025	0.538±0.024

6 Experiments

6.1 Setup

The experiments separate two tasks. The neural Cayley model evaluates anomaly-energy tracking under one shared connection ρ_t and cannot emit typed $\widehat{R}_{r,t}$. Typed operator identification is evaluated only by Procrustes and IT-GLLS on raw endpoint pairs; the domain injections therefore test the algebraic pair estimator, not typed recovery by the neural sheaf. I generate dynamic heterogeneous graphs (three node types, three edge types, 10 snapshots, stalk $d = 12$) and plant endpoint maps so that compliant pairs satisfy $x_v \approx R_t^\star x_u$ with $R_t^\star = (\rho^{\mathrm{tgt}})^+ \rho^{\mathrm{src}}$; 10% of targets are independent $\mathcal{N}(0, 2^2)$ anomalies. Smooth cells use Givens left-multiplies ($R_t^\star$ near $\mathrm{SO}(d)$); additive jumps can leave the group; a matched group jump stays in $\mathrm{SO}(d)$. Headline cells: slow/fast smooth, single jump at the split, and multi-jump. An adversarial-type cell (opposing Givens; not in Table 1) is used only for operator error (Table 4). Headline protocol: train on $t < 0.5$ only (no post-split features, labels, loss, or model-selection signal) and evaluate $t \geq 0.5$, with no planted jump indicator. The primary metric is post-drift edge AUROC. Elliptic++ (Weber et al., 2019; Elmisaad et al., 2023) has 165-dim features, illicit labels on a subset of transactions, and a documented market-structure change near timestep 43, but no ground-truth $R_t^\star$; I report a robustness–accuracy tradeoff, not relational-semantics identification. Baselines: MLP, TGN (Rossi et al., 2020), EvolveGCN-H (Pareja et al., 2020), static sheaf, DualPath (same routing, Euclidean fusion; GRU-only / Snap-only gates), Procrustes (Proposition 1), and IT-GLLS (Section 4.4). Same splits, positive class weight 9.0, at most 200 epochs.

6.2 Synthetic results

Table 1 is leak-free aggregated-node AUROC (Cayley sheaf, no pair residuals; five generator seeds). EvolveGCN and the sheaf are tied on slow drift (0.932±0.010 vs. 0.930±0.013); EvolveGCN leads on fast smooth (0.861±0.013). The sheaf still beats TGN, DualPath, and a static sheaf on slow drift, but jump cells overlap (sheaf 0.594±0.071, DualPath 0.590±0.062; Table 5). Typed operator recovery therefore uses pairs (Table 4): Procrustes RMS 0.042±0.035, IT-GLLS 0.001, while mixed and node-aggregated Procrustes stay near 0.25 (finite-sample SNR after mixed aggregation; Remark 1). The implemented Cayley coboundary does not emit a per-type $\widehat{R}$.

Table 2 separates observation from model class. Node-aggregated Procrustes is chance on all cells. Pair Procrustes solves smooth drift and the group jump, but fails the additive jump that leaves $\mathrm{SO}(d)$. Unlabeled IT-GLLS recovers both jump DGPs (five-seed AUC 1.000±0.000; Table 5). Aggregated Cayley and DualPath collapse at the split (0.594±0.071 vs. 0.590±0.062; Figure 3); opening DualPath's snapshot path does not recover the jump (Table 6). Oracle routing with planted jump labels reaches 0.845 and is not the unsupervised protocol. Routing comparisons are in Appendix E.

6.3 Elliptic++: robustness, not identification

Table 3: TGN and DualPath win overall illicit-edge AUC (0.872/0.867); the temporal sheaf is worst overall (0.646) but has the smallest pre/post degradation at the descriptive $t = 43$ cut (0.083 vs. 0.170–0.225). This is a robustness–accuracy tradeoff, not identification of $R_t^\star$ and not a win over dynamic GNNs on ranking quality. Native features do not support

Table 2: Identifiability of the residual (leak-free). Smooth Cayley is five-seed AUROC; additive/group jump columns are five-seed except pair Procrustes/IT-GLLS/oracle as previously logged. Additive jumps can leave SO(d); group jumps are Cayley left-multiplies. Pair methods see raw (x_u, x_v). IT-GLLS and r_e use no labels. Node-agg Procrustes is a finite-sample SNR check after mixed aggregation, not a corollary of Theorem 2(i).

Method	Observation	Smooth	Additive jump	Group jump
Node-agg Procrustes	aggregated nodes	0.522	0.492±0.009	0.547±0.021
Edge SO(d) Procrustes	pairs	0.991	0.460±0.147	0.998±0.003
Edge IT-GLLS (GL(d_x))	unlabeled pairs	1.000	1.000	1.000
Cayley sheaf (no pairs)	aggregated GNN	0.930±0.013	0.594±0.071	0.725±0.012
Two-map oracle	planted maps	1.000	1.000	1.000

Table 3: Elliptic++ chronological AUC, seed 42, no test-label adaptation. Train $t = 1$–34; test $t = 35$–49; degradation is pre minus post at the descriptive $t = 43$ cut. TGN node IDs are unique per transaction.

Model	Overall	Pre-shift	Post-shift (degr.)
TGN	**0.872**	0.848	0.670 (0.179)
DualPath	0.867	**0.865**	**0.678** (0.187)
Static sheaf	0.751	0.830	0.660 (0.170)
EvolveGCN-H	0.723	0.795	0.570 (0.225)
Temporal sheaf (mine)	0.646	0.684	0.602 (**0.083**)

$x_v \approx R_r x_u$, so IT-GLLS is not applied to illicit labels. The split trains on timesteps 1–34 and tests on 35–49; $t = 43$ is a cut inside the test window, not the train/test boundary. TGN uses unique transaction IDs (nodes do not persist across Elliptic snapshots); no model sees test labels.

On the same topology I plant a typed constraint in a 12-dimensional subspace (jump on `hub_to_leaf` at $t^\star$=30, before the market cut). IT-GLLS uniquely localizes $t^\star$ (rank 1/24, type-selective ratio 11.1:1); mixed Procrustes and EvolveGCN $\|\Delta W_t\|_F$ do not (Figure 2, Appendix D). The plant is synthetic; I do not claim native typed drift.

7 Discussion and limitations

The Cayley path improves smooth-drift scores (Table 1) but does not emit typed $\widehat{R}_r$. Unlabeled jump recovery needs pairs and GL(d_x); aggregated Cayley matches DualPath. Definition 2 is typed; the coboundary is a shared ρ_t. I do not claim that Elliptic++ is a relational-semantics-drift dataset, that Theorem 1 recovers ρ_t as a matrix, or that Cayley fusion recovers unlabeled jumps.

8 Conclusion

What can be recovered depends on what observation survives. Theorem 2(i) rules out unlinked mixed node-table marginals; raw typed pairs identify R_r (Propositions 1–2). A Cayley sheaf tracks smooth-drift coboundary energy near EvolveGCN (0.930±0.013 vs. 0.932±0.010), not a typed $\widehat{R}_r$; unlabeled jumps need pair IT-GLLS (AUC 1.000), not aggregated fusion (0.594±0.071). Elliptic++: TGN/DualPath lead overall AUC; the temporal sheaf only reduces $t = 43$ degradation, not identification of $R_t^\star$.

AI use statement

I used generative AI tools to draft and reorganize prose from existing internal research notes, to format LaTeX and BibTeX, and to check consistency between reported tables and JSON experimental logs. I used generative AI tools to help structure the paper against the ICLR 2027 author guidelines (page limits, required statements, double-blind constraints). I have not used generative AI tools to generate synthetic datasets, to invent experimental numbers, or to produce unverified mathematical claims: the architecture, theorems, and metrics come from my implementation and recorded runs. AI-assisted text was checked against source notes, code, and result files. I take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Ethics statement

This work uses public cryptocurrency transaction graphs (Elliptic / Elliptic++) and fully synthetic graphs with planted constraints. No private bank data or personally identifying customer records are used. The intended use is research on non-stationary graph learning and decision support for investigators, not fully automated freezing of accounts. A production AML system built on coboundary scores could amplify historical labeling bias (who was investigated) and could be gamed if adversaries approximate ρ_t. I recommend human-in-the-loop review, group-fairness monitoring of alert rates, and access controls on trained maps. The method does not provide legal determinations of criminal activity. I adhere to the ICLR Code of Ethics.

Reproducibility statement

Synthetic generator, dual-path Euclidean baseline, model code (V3.5 dual-path sheaf), TGN and EvolveGCN baseline scripts, adversarial-type, identifiability-boundary, and operator-recovery experiments, and JSON result files are included in the anonymous supplementary material. Assumption statements and the derivation of the informal budget Theorem 1 and the proof of Theorem 2 appear in Section 5 and Appendices A–B. Lemma 1 is a standard OLS rate (Vershynin, 2018; Wainwright, 2019); I do not prove a trimming guarantee for IT-GLLS. Default hyperparameters ($d = 12$, hidden 128, three diffusion layers, curriculum 80/40/120, positive weight 9.0) are listed in Appendix C. Elliptic++ processing (feature columns, chronological split, regime index 43) is described in Appendix D. Headline synthetic numbers in Table 1 use five-seed leak-free temporal holdout on the aggregated-node Cayley sheaf: no post-split features, labels, task loss, or model-selection information, and no planted jump indicators. Table 6 routing cells remain the seed-42 DualPath-mode run. Table 2 and Table 5 add unlabeled pair-level IT-GLLS. Table 4 is RMS operator error on the adversarial-type cell (supplementary JSON). Table 7 isolates the oracle-routing diagnostic (0.845 jump) that uses planted jump indicators.

References

Md Sadek Hossain Asif, Tanzila Khan, and Md. Mosaddek Khan. Temporal sheaf neural networks with dynamic orthogonal transport. arXiv preprint arXiv:2606.10071, 2026.

Federico Barbero, Cristian Bodnar, Haitz Sáez de Ocáriz Borde, Michael Bronstein, Petar Veličković, and Pietro Lió. Sheaf neural networks with connection Laplacians. In ICML Workshop on Topology, Algebra, and Geometry in Machine Learning, 2022.

Albert Bifet and Ricard Gavaldà. Learning from time-changing data with adaptive windowing. In Proceedings of the SIAM International Conference on Data Mining, 2007.

Cristian Bodnar, Francesco Di Giovanni, Benjamin Paul Chamberlain, Pietro Lió, and Michael M. Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing. In Advances in Neural Information Processing Systems, 2022.

Mário Cardoso, Pedro Saleiro, and Pedro Bizarro. LaundroGraph: Self-supervised graph representation learning for anti-money laundering. In Proceedings of the 3rd ACM International Conference on AI in Finance, 2022.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In Advances in Neural Information Processing Systems, 2018.

Youssef Elmisaad, Jiahao Liu, Adam Nassar, and Evangelos Papalexakis. Elliptic++: A dataset and benchmark for cryptocurrency transaction graphs. arXiv preprint arXiv:2306.16424, 2023.

Jean Gallier. Geometric Methods and Applications: For Computer Science and Engineering. Springer, 2 edition, 2011.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. ACM Computing Surveys, 46(4), 2014.

Jakob Hansen and Thomas Gebhart. Sheaf neural networks. In NeurIPS Workshop on Topological Data Analysis and Beyond, 2020.

Jakob Hansen and Robert Ghrist. Toward a spectral theory of cellular sheaves. Journal of Applied and Computational Topology, 3:315–358, 2019.

Calypso Herrera, Florian Krach, Pierre Ruyssen, and Josef Teichmann. Neural jump ordinary differential equations. In International Conference on Learning Representations, 2021.

Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. Heterogeneous graph transformer. In Proceedings of the Web Conference, 2020.

Seyed Mehran Kazemi, Rishab Goel, Kshitij Jain, Ivan Kobyzev, Akshay Sethi, Peter Forsyth, and Pascal Poupart. Representation learning for dynamic graphs: A survey. In Journal of Machine Learning Research, 2020.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In Advances in Neural Information Processing Systems, 2020.

Edgar Alonso Lopez-Rojas, Ahmad Elmir, and Stefan Axelsson. PaySim: A financial mobile money simulator for fraud detection. European Modeling and Simulation Symposium, 2016.

Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao B. Schardl, and Charles E. Leiserson. EvolveGCN: Evolving graph convolutional networks for dynamic graphs. In Proceedings of the AAAI Conference on Artificial Intelligence, 2020.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. Neural Networks, 113:54–71, 2019.

Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. In ICML Workshop on Graph Representation Learning and Beyond, 2020.

Yulia Rubanova, Ricky T. Q. Chen, and David Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In Advances in Neural Information Processing Systems, 2019.

Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In European Semantic Web Conference, 2018.

Joakim Skarding, Bogdan Gabrys, and Katarzyna Musial. Foundations and modelling of dynamic networks using dynamic graph neural networks: A survey. IEEE Access, 9, 2021.

Roman Vershynin. High-Dimensional Probability: An Introduction with Applications in Data Science. Cambridge University Press, 2018.

Martin J. Wainwright. High-Dimensional Statistics: A Non-Asymptotic Viewpoint. Cambridge University Press, 2019.

Mark Weber, Jie Chen, Toyotaro Suzumura, Aldo Pareja, Tengfei Ma, Hiroki Kanezashi, Tim Kaler, Charles E. Leiserson, and Tao B. Schardl. Scalable graph learning for anti-money laundering: A first look. In NeurIPS Workshop on Challenges and Opportunities for AI in Financial Services, 2018.

Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I. Weidele, Claudio Bellei, Tom Robinson, and Charles E. Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. In KDD Workshop on Anomaly Detection in Finance, 2019.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. In International Conference on Learning Representations, 2020.

Jiaxuan You, Tianyu Du, and Jure Leskovec. ROLAND: Graph learning framework for dynamic graphs. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022.

Le Yu, Leilei Sun, Bowen Du, and Chuanren Liu. Towards better dynamic graph learning: New benchmark and weakly supervised dynamic graph neural network. In Advances in Neural Information Processing Systems, 2023.

A Derivation of Theorem 1

The argument is a structural error budget for energies, not a uniform high-probability theorem for matrices. Write $E_t(\rho) = \frac{1}{n_t}\sum_e \|(\delta[\rho]\, s_t)(e)\|^2$. The estimator $\widehat{\rho}_t$ is the connection used by the detector; I do not assume it is an empirical-risk minimizer over a class $\mathcal{H}$.

Step 1: energy identification. Assumption 1 implies that the conditional label law identifies coboundary energy through g, not $\rho \in \mathrm{SO}(d)$. Local Lipschitz invertibility of g transfers label-space error into energy-space error when one inverts g at a representative energy. $\dim \mathrm{SO}(d) = d(d-1)/2$ (here 66), so a scalar E_t cannot have a unique minimizer on an open set of $\mathrm{SO}(d)$. That is why equation 9 is stated for $|E_t(\widehat{\rho}_t) - E_t^\star|$.

Step 2: local estimation of the energy. Fix a window $W_t = [t, t+\Delta_t]$ containing no jump. Under Assumption 3 the empirical energy at a fixed ρ has variance $O(1/n_t)$, hence a $C_1 n_t^{-1/2}$ term. This is pointwise in ρ. Controlling a data-dependent $\widehat{\rho}_t \in \mathcal{H} \subset \mathrm{SO}(d)$ would require uniform concentration (covering numbers of $\mathcal{H}$), which I do not prove. Binary labels concentrate around $\frac{1}{n_t}\sum_e g(\|(\delta^\star s_t)(e)\|^2)$, not around $g(E_t^\star)$. For nonlinear g this leaves a Jensen remainder $\left|\frac{1}{n_t}\sum_e g(E_e) - g(E_t^\star)\right|$ that need not vanish if energies are heterogeneous across the window. I absorb that remainder, together with ordinary label noise, into the informal $C_4\varepsilon_{\mathrm{label}}$ slot rather than displaying it separately.

Step 3: drift bias. If $\rho^\star$ is L_{drift}-Lipschitz on W_t, a constant-in-window energy estimate incurs bias $O(\Delta_t L_{\mathrm{drift}})$ because coboundary energy is locally Lipschitz in ρ for bounded stalks. This is the C_2 term and is undefined across a jump.

Step 4: model class. Let $\mathcal{H}$ be any parameterized family and set $\varepsilon_{\mathrm{model}}$ to the energy gap between $\rho^\star$ and the best element of $\mathcal{H}$ on W_t. This is an approximation gap, not an optimization-error term for the non-convex curriculum, and not a bound on jerk-gate false positives or false negatives. I do not claim that a GRU equals a neural ODE, nor that Cayley charts universally approximate $\mathrm{SO}(d)$.

Step 5: stitching. A smoothness penalty averages window estimators. A union bound over $O(T/\Delta_t)$ windows would contribute a $\log(T/\Delta_t)$ factor to any high-probability statement; I do not display it because equation 9 is not claimed as a uniform tail bound.

When ρ is identifiable. Proposition 1 restores uniqueness on $\mathrm{SO}(d)$ if the population criterion is the matrix $\mathbb{E}[x_v x_u^\top]$, $R^\star \in \mathrm{SO}(d)$, and $\mathbb{E}[\xi x_u^\top] = 0$. Proposition 2 identifies $R^\star \in \mathrm{GL}(d)$ by unconstrained least squares under the same exogeneity; Lemma 1 records the standard OLS rate. Theorem 2(i) is the complementary negative for the product of mixed-type marginals. Table 2 is a finite-sample SNR check on linked mixed pairs.

At a jump, equation 9 is undefined. I do not prove a hidden-state mismatch bound, a curriculum optimization bound, or a guarantee for the heuristic jerk detector; Figure 3 is the empirical check.

B Proof of Theorem 2

A random vector $z \in \mathbb{R}^{d_x}$ is isotropic if $Oz \overset{d}{=} z$ for every $O \in \mathrm{SO}(d_x)$. Standard Gaussians $\mathcal{N}(0, \sigma^2 I)$ are isotropic; the generator draws sources and anomalous targets from this class.

Part (i). Fix a node w. Each incident raw endpoint $z \in I(w)$ is of one of three kinds: (a) a source x_u^e, isotropic by Assumption 4; (b) a compliant target $R_r x_u^e + \xi^e$. Isotropy of x_u^e gives $R_r x_u^e \overset{d}{=} x_u^e$, and adding independent isotropic ξ^e preserves isotropy with a law that does not depend on R_r; (c) an anomalous target, isotropic and independent of $\mathbf{R}$. Distinct edges have independent sources, and a non-loop edge contributes at most one endpoint to w, so the summands in equation 11 are independent and each has a law that does not depend on $\mathbf{R}$. Therefore $\bar{x}_w$ has a law that does not depend on $\mathbf{R}$. The product of marginals $\prod_w \mathbb{P}(\bar{x}_w)$ is consequently constant on $\mathrm{SO}(d_x)^R$. Any two distinct tuples $\mathbf{R} \neq \mathbf{R}'$ (e.g. a transposition of two unequal type maps) induce the same product of node-table marginals, so the map is not injective.

Part (ii). On compliant type-r pairs, $\mathbb{E}[x_v^e (x_u^e)^\top \mid r, \text{ compliant}] = \mathbb{E}[(R_r x_u^e + \xi^e)(x_u^e)^\top \mid r] = R_r \Sigma_r$ because $\mathbb{E}[\xi^e (x_u^e)^\top] = 0$ by independence and $\mathbb{E}[\xi^e] = 0$ (isotropy about the origin). If a fraction α of type-r targets are independent isotropic anomalies, the unconditional moment is $(1-\alpha) R_r \Sigma_r$; the polar factor on $\mathrm{SO}(d_x)$ is unchanged. If Σ_r is full rank, Proposition 1 says that the unique Procrustes minimizer on $\mathrm{SO}(d_x)$ is the polar factor of S_r, which equals R_r. If $|N_r(u)| = 1$, the node-wise outer product $x_v x_u^\top$ has rank at most 1. The stabilizer in $\mathrm{SO}(d_x)$ of a nonzero vector is isomorphic to $\mathrm{SO}(d_x - 1)$ and has dimension $(d_x - 1)(d_x - 2)/2$. For $d_x \geq 3$ that dimension is positive, so a rank-1 constraint does not separate maps on an open set of $\mathrm{SO}(d_x)$. For $d_x = 2$ the stabilizer is discrete: a single nonzero pair $x_v = R x_u$ determines the rotation angle, so the rank-1 obstruction does not apply.

What is not claimed. The argument never uses Fisher information of stacked $\{\bar{x}_w\}$: $n d_x$ coordinates can exceed $\dim \mathrm{SO}(d_x)^R$, so a rank count on the concatenated node table would not prove non-identifiability. Part (i) is about the product of unlinked marginals. On an edge $e = u \to v$ of compliant type r, the mixed averages still contain the original summands x_u^e and $R_r x_u^e + \xi^e$, so $\mathbb{E}[\bar{x}_v \bar{x}_u^\top \mid e \in E_r, \text{ compliant}] \propto R_r \Sigma_r$ with a factor of order $\mathbb{E}[1/(|I(u)||I(v)|)]$. That linked mixed moment is a degree-diluted copy of S_r, a sample-complexity issue rather than population non-identifiability. Empirically polar of the mixed cross-moment does not recover R_r at the sample sizes used here (Table 2, node-agg column). Per-relation aggregation computes a different statistic than equation 11 and is outside the theorem. If a persistent node feature is reused across incident edges, the independence step in Part (i) fails and typed maps can in principle leak into node-level joints; that DGP is outside Assumption 4 (Remark 2).

C Implementation details

Architecture. Stalk dimension $d = 12$; raw features $d_x = 16$ (synthetic) or $d_x = 165$ (Elliptic++); GRU hidden size 128 with 2 layers; 3 diffusion layers; 3 edge types; dropout 0.1; Cayley $\varepsilon = 10^{-8}$. Jerk EMA $\alpha = 0.3$, relative threshold $\tau_{\text{jerk}} = 2.0$, recovery window $W = 6$, slow-gate floor 0.1 (MLP on $[h_t; c_t^{\text{old}}]$, final bias +2), fusion-gate EMA $\beta = 0.9$ with g^{emb} affine init $(w, b) = (5, -2)$ as in equation 5. Positive class weight 9.0. Stalk projection is $s_t = Px_t$ with $P \in \mathbb{R}^{d \times d_x}$ learned. The edge head is $(2d[+1]) \to 64 \to 32 \to 1$ (ReLU, dropout 0.1); the extra coordinate is r_e when pairs are observed.

Cayley chart. I shrink fast-encoder Lie coefficients by 0.1 before Cayley$(\cdot)$ and add εI to the Cayley linear solve. This keeps ρ_t^{fast} near the identity. It does not keep $(\rho_t^{\text{slow}})^\top \rho_t^{\text{fast}}$ away from eigenvalue -1 after ρ_t^{slow} has drifted, and it limits large group jumps proposed by the snapshot path. I do not monitor $\lambda_{\min}((\rho_t^{\text{slow}})^\top \rho_t^{\text{fast}} + I)$.

Diffusion operator. Write M_ρ for the (undirected) aggregation of restriction-map messages $s_u\rho$ onto targets and $s_v\rho^\top$ onto sources, and D for the corresponding degrees (clamped at 1). Each layer is $X^{(l+1)} = \text{LN}\big(\text{Dropout}(\sigma(W^{(l)}(X^{(l)} - D^{-1}M_\rho X^{(l)}))) + X^{(l)}\big)$ with $\sigma = \text{ReLU}$. This is a degree-normalized unit-step restriction-map diffusion, not $(I - \eta\delta^\top\delta)XW$ for a free η. I therefore do not report an η that would stabilize an unnormalized sheaf Laplacian on high-degree hubs.

Optimization. Adam; phase-1/2/3 learning rates 10^{-3}, gate $5 \cdot 10^{-3}$, fine-tune $3 \cdot 10^{-4}$; gradient clip 1.0. Loss weights: $\lambda_{\text{s}} = 0.005$, $\lambda_{\text{f}} = 0.1$, $\lambda_o = 0.01$, $\lambda_{\text{j}} = 0$ (unsupervised) or 1 (oracle). $\mathcal{L}_{\text{fast}}$ is the mean coboundary norm under ρ_t^{fast}, matching the implemented detector statistic. Curriculum 80/40/120 epochs on synthetic data with jump labels; the unsupervised protocol is 80/0/160 with training restricted to $t < 0.5$. Elliptic++ has no jump labels and therefore uses 60/0/120; the Phase 2 budget is moved into Phase 3. Its static pretrain is 200 epochs.

Scoring. Edge AUROC on query edges of the current snapshot. No future edges are visible at time t. When pairs are available, IT-GLLS residuals are concatenated to the edge head (Section 4.4); residual-only scoring is also reported.

Compute. Synthetic runs complete on CPU; Elliptic++ uses a single GPU when available. I did not tune baselines beyond published defaults adapted to my splits.

Curriculum. Phase 1 freezes the GRU, forces $g^{\text{fuse}} = 1$, and trains ρ^{fast} and the scoring head. Phase 2 (optional) freezes encoders and trains the change detector with jump BCE; skipped when jump labels are withheld. Phase 3 is end-to-end fine-tuning at a lower learning rate. The unsupervised protocol moves the Phase 2 budget into Phase 3.

Complexity. Let $n = |V_t|$, $m = |E_t|$, stalk size d, ambient feature size d_x, and $R = |\mathcal{T}_E|$. Coboundary evaluation is $O(md^2)$. Each Cayley evaluation is $O(d^3)$ ($d = 12$ in all experiments). The snapshot encoder is one round of mean aggregation, $O(md_h + nd_h^2)$. GRU cost is independent of n because it consumes a graph-level coboundary summary. IT-GLLS is $O(Rmd_x^2)$ per snapshot when raw pairs in $\mathbb{R}^{d_x}$ are observed ($d_x = 16$ synthetic; $d_x = 12$ on the projected Elliptic inject).

D Data processing

Synthetic generator. For each edge type r, sample endpoint maps $(\rho_{r,0}^{\text{src}}, \rho_{r,0}^{\text{tgt}})$ by Cayley of random skew matrices. Smooth scenarios apply Givens left-multiplies. Jump scenarios are additive by default: $\rho \leftarrow \rho + \text{magnitude} \cdot \Delta$ ($\|\Delta\|_F = 1$), which can take $R = (\rho^{\text{tgt}})^+ \rho^{\text{src}}$ out of $\text{SO}(d)$. A matched group jump left-multiplies by Cayley$(A) \in \text{SO}(d)$. Compliant targets are regenerated from the current maps; a frozen 10% of edges have independent anomalous

targets $\mathcal{N}(0, 2^2)$. Raw endpoint features are then collapsed onto nodes by mixed-type mean aggregation equation 11 ($d_x = 16$); Theorem 2 applies to that node table. Moderate/extreme regimes increase isotropic noise and flip labels with probabilities 0.05 and 0.20.

Elliptic++. I use official timestep indices 1–49, node features of dimension 165, and binary illicit labels where available (unknowns excluded from the loss). Train on timesteps before the evaluation window listed in the supplementary JSON; report AUC per test timestep and average. A documented market-structure change occurs near timestep 43 (Weber et al., 2019); I use it as a chronological cut, not as ground-truth relational-semantics drift. Several late timesteps have very few labeled illicit nodes, which produces 0.5 AUC for both models; I include them in the mean rather than filtering.

Elliptic inject protocol. To test type-selective detection on real topology without attributing the t=43 market shift to ρ_t, I plant a single-type constraint in a 12-dimensional projected subspace of the real 165-dim features (degree-ratio edge types from topology only; no label leakage). Compliant pairs obey $x_v^{\text{proj}} \approx R_r(t) x_u^{\text{proj}}$; R_r drifts smoothly, and only `hub_to_leaf` receives an additive jump at $t^\star$=30. The null space of the projection is left unchanged. Typed maps are unlabeled IT-GLLS on the projected pairs, with carry-forward on snapshots whose typed source Gram is ill-conditioned. The implemented Cayley coboundary is still a single ρ and is not scored for per-type Δ_r. IT-GLLS $\Delta_{\texttt{hub_to_leaf}}(t^\star)$ is the unique maximum among 24 eval timesteps (rank 1/24), with type-selective ratio 11.1:1 over unaffected types at $t^\star$. Mixed-type Procrustes and EvolveGCN $\|\Delta W_t\|_F$ do not localize $t^\star$ (ranks 23 and 19). Orthogonal polar of S_r does not uniquely peak: the plant is additive and can leave SO(d).

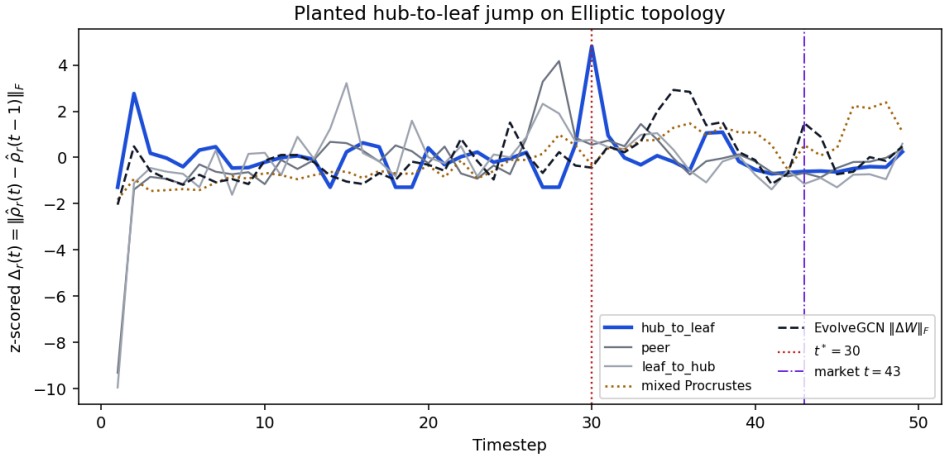

Figure 2: Elliptic topology inject. Planted `hub_to_leaf` jump at $t^\star$=30 on real Elliptic topology. Series are z-scored per method. Typed IT-GLLS spikes at $t^\star$; mixed Procrustes and EvolveGCN do not isolate the type. The market-structure change at t=43 is a separate chronological cut.

IBM HI-Small inject (same protocol, payment topology). I repeat the plant on the public IBM HI-Small transfer graph (calendar days 2022-09-01–10; days with fewer than 10,000 payments dropped). Edge types are payment kinds; random account features carry a 12-dimensional map $x_v \approx R_r(t) x_u$; only `ACH` receives an additive jump on 2022-09-08. Amounts and counterparties are unchanged. IT-GLLS $\Delta_{\texttt{ACH}}(t^\star)$ is the unique maximum among five eval days (type-selective ratio 12.8:1); day-to-day amount volume, per-sender totals, and new-pair rate do not uniquely peak. This is the same planted identification check as Figure 2 on a bank-shaped graph, not native IBM laundering labels and not a claim that the simulator exhibits relational-semantics drift.

Table 4: RMS operator error on the adversarial-type smooth cell (opposing Givens; five seeds; leak-free $t \geq 0.5$), not the default slow-smooth row of Table 1. Error is $\sqrt{\text{mean}((\widehat{R} - R^\star)^2)}$ (same helper as prior 0.042 citations; not Frobenius). Procrustes vs. polar($R^\star$); IT-GLLS vs. unpolarised $R^\star = (\rho^{\text{tgt}})^+ \rho^{\text{src}}$ (vs. polar: 0.142±0.033). Em-dashes: no evaluable per-type relative map.

Method	Observation	Emits typed $\widehat{R}$?	RMS error
Typed Procrustes	raw pairs	yes	0.042±0.035
Mixed Procrustes	raw pairs (one map)	no	0.261±0.016
Node-agg Procrustes	mixed node table	misspecified	0.248±0.011
IT-GLLS (GL(d_x))	unlabeled pairs	yes (in GL)	0.001±0.000
Cayley sheaf	aggregated GNN	—	—
EvolveGCN	aggregated GNN	—	—
TGN	aggregated GNN	—	—
DualPath	aggregated GNN	—	—

Table 5: Leak-free clean single-jump post-AUC over five generator seeds. Sheaf/DualPath are aggregated-node models. IT-GLLS and V3.5+r_e observe raw pairs. Mean±std uses Bessel's correction.

	42	43	44	45	46	Mean±std
Sheaf	0.655	0.483	0.586	0.658	0.590	0.594±0.071
DualPath	0.619	0.485	0.592	0.645	0.608	0.590±0.062
IT-GLLS / V3.5+r_e	1.000	1.000	1.000	1.000	1.000	1.000±0.000

Protein contact-graph inject (Test P-A; mechanism only). I repeat the plant on public crystal 1UBQ chain A (76 residues). Edge types are sequence-separation plus Cα distance: backbone ($|i-j|$=1), short-range ($2 \leq |i-j| \leq 4$, d<8 Å), long-range ($|i-j| \geq 5$, d<8 Å), distal ($|i-j| \geq 5$, $8 \leq d$<12 Å). Random residue features carry a 12-dimensional map $x_v \approx R_r(t) x_u$; only long-range receives an additive jump at frame $t^\star$=7. Cα coordinates receive thermal noise only. IT-GLLS $\Delta_{\texttt{long_range}}(t^\star)$ is the unique maximum among seven eval frames (type-selective ratio 7.5:1) and the argmax type at $t^\star$ is long-range. Frame-to-frame Cα RMSD, RMSF, contact count, new-contact rate, and mean residue motion do not uniquely peak. Native Cα endpoints and native backbone frames do not obey a usable shared linear pair law on non-backbone contacts (Gate 0 fail: held-out frame residual is at chance vs. a destination shuffle); the plant is therefore a synthetic subspace, as in the IBM inject. This is not a ligand- or mutant-induced transition, not allostery, and not a drug-design result. Typed shared R_r on tertiary contacts is treated as a plant-only object for this application branch (Gate 0 abandoned there). The AdK hinge-highest criterion is withdrawn: leftover is a pair-local strain sensor, not a hinge detector. On KRAS G12C (GDP 4L8G vs AMG 510 6OIM) pair-local leftover is higher on Switch I–II and SII-P contacts than on CORE, but it does not localize the cryptic pocket better than global Cα displacement or pairwise |Δdistance| (displacement enrichment 1.98 vs leftover 1.41; Spearman leftover vs $|\Delta d|$ 0.75). That is not a drug-ranking result (P-D unopened).

E Additional experimental notes

Operator recovery and jump curves. Table 4 is RMS operator error on the adversarial-type smooth cell (opposing Givens). Table 5 is the five-seed leak-free jump comparison. Figure 3 is per-timestep AUC around the planted split.

What the baselines cannot do. TGN's memory tracks node histories under a fixed relation operator. EvolveGCN evolves one global W_t in ambient matrix space. DualPath adds the routing skeleton but fuses Euclidean embeddings: opening its snapshot path still fails the aggregated jump (Table 6). A static sheaf is a single-regime lower bound after $\rho^\star$ leaves the training window. None of these models observes raw pairs, so none can run IT-GLLS.

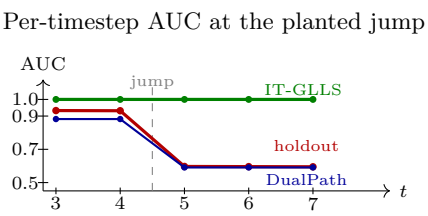

Figure 3: Jump-time AUC. Per-timestep AUC around the planted jump at $t = 0.5$. Holdout Cayley sheaf and DualPath are five-seed aggregated-node means; IT-GLLS is unlabeled pair residual (also 1.0 for V3.5+r_e). Aggregated Cayley does not recover; pair GL(d) does.

Operating regimes and non-claims. When drift is Lipschitz and $R_t^\star$ stays near SO(d), the slow Cayley path improves anomaly prediction under the planted smooth process (Table 1); it does not emit a typed $\widehat{R}_r$. When a jump takes $R_t^\star$ out of SO(d), pair Procrustes fails and aggregated Cayley fusion does not recover (0.594±0.071; Figure 3). When the jump stays on SO(d), pair Procrustes recovers (0.998±0.003) but node-aggregated Procrustes remains near chance (0.547±0.021): the practical blocker is SNR after mixed aggregation, not the fusion gate (Remark 1). S_r is the identifying statistic that restores recovery; it is a pooled summary of raw pairs, not a TGN node table. Matched group jumps raise leak-free aggregated Cayley only to 0.725±0.012 (single jump) and 0.687±0.022 (multi-jump), still far from the pair estimator. On the same group multi-jump DGP, DualPath is 0.723±0.025 vs. Cayley 0.687±0.022 (Table 6). Thus membership of $R_t^\star$ in SO(d) alone does not make Cayley fusion superior; chart proximity, matrix-solve gradients, and routing are possible explanations, but I did not log the diagnostics needed to distinguish them. Unlabeled recovery requires endpoint pairs and a GL(d) residual (Table 2). Oracle routing with planted jump labels reaches 0.845 (Table 7) but is not the unsupervised protocol. When jumps are frequent and pairs are unobserved, $\varepsilon_{\text{model}}$ dominates; multi-jump and extreme-noise cells show this. Native Elliptic++ features do not satisfy the typed linear pair model, so the residual of Section 4.4 is applied only on the planted-subspace inject of Figure 2, not to illicit labels. The gap between the typed formulation of Theorem 2 and the single-ρ implementation reflects the paper's structure: Theorem 2 characterizes what observation is needed for typed identification, and the pair estimator (IT-GLLS) achieves it empirically. The neural Cayley path demonstrates that SO(d) energy tracking is feasible for smooth drift under a shared connection; per-type neural tracking requires either per-type message passing or a multi-head Cayley parameterization and is future work. A large jump gap versus TGN/EvolveGCN on the aggregated protocol shows those models fail a phenomenon they were not designed for; it does not show that Cayley geometry recovers the jump. The fair evidence for typed-map recovery on smooth drift is Table 4, not a detection margin versus EvolveGCN in Table 1. I do not identify causal regulatory events, claim universal AML typology coverage, or assert adversarial robustness. I do not claim Theorem 1 as matrix recovery of ρ_t on SO(d), nor as a jump-time guarantee or a justification of the GRU. I do not claim that V3.5+r_e is more than a wrapper around the pair residual. I do not report a neural sheaf operator error in Table 4: the implemented coboundary applies a single ρ. The supported real-data claim is a robustness–accuracy tradeoff under one chronological split.

Routing ablations and oracle routing. Table 6 holds the DualPath skeleton fixed (GRU-only / Snap-only / DualPath vs. sheaf). Table 7 isolates the 0.845 oracle-routing diagnostic from the leak-free holdout. Additive multi-jump DualPath is 0.531±0.025 (Table 1); group multi-jump DualPath is a different DGP and is reported on its own row.

Why identity reset fails. On clean single-jump graphs, forcing $\rho \leftarrow I$ at the detected jump yields post-drift AUC in $[0.65, 0.70]$. The identity is the wrong constraint: coboundary energy under I does not reproduce the planted violation law. A Cayley snapshot encoder is also the wrong class after an additive jump. Unlabeled recovery uses the GL(d) residual of Section 4.4, not a group-valued reset.

Table 6: Routing versus geometry, leak-free aggregated-node protocol (post-drift AUROC). DualPath is GRU+snapshot+gate with Euclidean fusion. GRU-only forces $g = 0$; Snap-only forces $g = 1$. Sheaf column is Cayley without pair residuals. Additive rows are seed 42 from the DualPath-mode script (not the five-seed dual-only grid in Table 1). Group multi-jump is five-seed mean on Cayley left-multiply ($R^\star$ stays in SO(d)).

Setting	GRU-only	Snap-only	DualPath	Sheaf (mine)
Slow smooth	0.876	0.849	0.856	**0.940**
Fast smooth	0.828	0.803	0.809	**0.843**
Single jump	0.614	0.612	0.621	0.656
Multi-jump (additive)	0.518	0.512	0.525	0.546
Multi-jump (group)	0.724	0.709	0.723	0.687

Table 7: Oracle routing vs. leak-free holdout (clean post-drift AUROC). Oracle uses jump BCE, GT hard-handoff, and Phase-1 task loss on post-split snapshots. Holdout is aggregated Cayley, not IT-GLLS.

Setting	Oracle routing	Holdout (mine)	DualPath	Static
Slow smooth	0.976	0.940	0.856	0.786
Fast smooth	0.915	0.843	0.809	0.710
Single jump	0.845	0.656	0.621	0.604
Multi-jump	0.672	0.546	0.525	0.524

Euclidean DualPath protocol. The DualPath baseline uses a two-layer mean-aggregation GNN, a GRU over graph embeddings, an embedding-shift gate, and convex fusion in $\mathbb{R}^h$. It has no coboundary, no Cayley map, and no three-phase curriculum (those are sheaf-specific). On additive single-jump seed 42, GRU-only / Snap-only / DualPath obtain 0.614 / 0.612 / 0.621. The DualPath gate mean is 0.15 (mostly closed), but Snap-only forces $g = 1$ and still fails (0.612). On matched group multi-jump (five seeds), DualPath is 0.723±0.025 with gate mean 0.17; GRU-only is 0.724±0.026. That DualPath number is not comparable to additive DualPath 0.525. The 0.845 sheaf jump score uses planted jump indicators; leak-free aggregated sheaf is 0.594±0.071 over five seeds (Table 5). The snapshot DualPath path is a supervised embedding scorer trained only before the split; the 0.845 sheaf run additionally used Phase 1 task loss on post-split snapshots.

Inconclusive GL(d) relabel. Replacing sheaf labels by a jumping invertible affine rule $A_t x_u \approx x_v$ on the same sheaf-generated features did not reverse rankings: DualPath GRU post-AUC was 0.978 while Snap-only was 0.500. That relabel is distinct from the additive jump, which already takes the planted relative map out of SO(d). I therefore do not use the relabel as a mismatched-DGP stress test.

Routing collapse. End-to-end training without the curriculum often leaves $g^{\text{fuse}} \approx 0$, reducing the model to the slow path and reproducing the 0.65 jump failure of the aggregated GNN. Phase 1 forces the fast path to be a competent expert before the gate is allowed to choose.

Ceiling. A privileged model that receives planted $(\rho^{\text{src}}, \rho^{\text{tgt}})$ on test snapshots defines an empirical upper bound of 1.000 (two-map oracle). Unlabeled IT-GLLS on raw pairs matches that bound. On clean slow drift, oracle-routing post-AUC (0.976) matches the previous Cayley ceiling (0.975) within noise.

F Pseudocode

Table 8: What is allowed to evolve.

Family	Evolves	Held fixed
Temporal GNNs	Node / edge state	Relation operators
EvolveGCN / ROLAND	GNN weights	Constraint geometry
DualPath (this paper)	Embeddings + gate	Operator manifold
R-GCN / HGT	(typically nothing)	Type maps
This work	ρ_t ($R_t = \rho_t^2$); pair GL(d)	Sheaf axioms

Algorithm 1 One temporal step of dual-path sheaf inference

Require: snapshot $(\mathcal{G}_t, x_t)$, persistent (ρ_{t-1}, h_{t-1}), protocol $\in \{\text{unsup}, \text{oracle}\}$
$s_t \leftarrow P x_t$ ▷ learned $P: \mathbb{R}^{d_x} \to \mathbb{R}^d$
$c_t^{\text{old}} \leftarrow \|\delta[\rho_{t-1}]\, s_t\|$; update jerk detector; $g_t^{\text{rec}} \leftarrow$ recovery step
$\rho_t^{\text{slow}}, h_t \leftarrow \text{GRU-Cayley}(h_{t-1}, c_t^{\text{old}}, \rho_{t-1})$ ▷ equation 4
$\rho_t^{\text{fast}}, z_t \leftarrow \text{SnapshotCay}(\mathcal{G}_t, x_t)$
$g_t^{\text{emb}} \leftarrow$ cosine gate equation 5; $g_t^{\text{fuse}} \leftarrow \max(g_t^{\text{emb}}, g_t^{\text{rec}})$
if oracle and planted jump then $g_t^{\text{fuse}} \leftarrow 1$; $\rho_t \leftarrow \rho_t^{\text{fast}}$; reset GRU from ρ_t^{fast}
else $\rho_t \leftarrow$ Cayley interpolate equation 6
end if
if pairs observed then fit $C_{r,t} \in \text{GL}(d_x)$ by IT-GLLS; $r_e \leftarrow \|x_v - C_{r,t} x_u\|$
end if
diffuse s_t with degree-normalized M_{ρ_t}; score edges (optionally with r_e)
persist ρ_t (after oracle handoff, persist ρ_t^{fast}) as ρ_t for the next step
return $\rho_t, h_t, \hat{p}_t$