HIERARCHICAL GRANGER-CAUSAL ROOT CAUSE ANALYSIS OF ANOMALIES IN INDUSTRIAL MULTIVARIATE TIME SERIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Root cause analysis (RCA) of industrial multivariate time series aims to identify the variables directly responsible for anomalous events. This task is challenging because disturbances can propagate across interacting subsystems, causing many sensors to respond to a small number of root variables. Existing methods mainly model dependencies at the variable level, which may mix subsystem-level dynamics with local variable interactions. We propose HiRCA, a hierarchical Granger-causal predictive framework that models these two levels separately. HiRCA uses a soft engineering prior on subsystem membership to learn latent subsystem states and their temporal dependencies. At the variable level, it separates the effects of self-history, shared subsystem context, and cross-variable interactions on prediction. For diagnosis, HiRCA first aggregates calibrated innovation scores to identify candidate subsystems and then prioritizes their variables in the final ranking. On two synthetic benchmarks, HiRCA shows strong hierarchical graph-recovery performance and achieves Avg@10 scores above 0.90 for root cause localization. Across four real-world datasets, HiRCA ranks among the top three in nearly all evaluations, including 16 top-two variable ranking results and 15 best time-variable ranking results. These results demonstrate the effectiveness of hierarchical predictive modeling for structured, coarse-to-fine root cause localization in industrial multivariate time series.

1 INTRODUCTION

Industrial multivariate time series record interacting sensor measurements during normal and anomalous operation. Root cause analysis (RCA) aims to identify the variables that are directly perturbed during an anomaly. Disturbances can propagate through interacting process dynamics and appear across many sensors, so an observed anomaly may contain both direct and propagated responses (Qin, 2012). This distinction is important for diagnosis. Downstream alarms indicate where a disturbance appears, whereas root variables indicate where corrective action should begin. Effective RCA therefore needs to distinguish underlying perturbations from their propagated responses (Venkatasubramanian et al., 2003). Causal reasoning is useful in this setting because it makes fault propagation paths explicit and can improve diagnostic interpretability in industrial systems (Li et al., 2025a).

As shown in Figure 1, industrial multivariate time series have structure above the variable level. Sensors are often grouped into components or process units, and variables in the same subsystem share common dynamics. When a disturbance propagates between subsystems, it may affect several variables on both sides. A flat variable graph may then represent a single subsystem-level propagation event with many variable-level edges. Within a subsystem, shared variation can also obscure the additional predictive contribution of an individual variable. A flat diagnostic rule further ranks all variables together, even when abnormal evidence is concentrated in only a few subsystems.

Most existing RCA methods reason over variables at a single resolution (Chen et al., 2018; Han et al., 2025; Lv et al., 2025), while prior hierarchical approaches do not directly separate learned subsystem dynamics from conditional variable-level effects under an engineering membership prior. We therefore use the engineering membership map as a soft prior and learn the subsystem states

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.

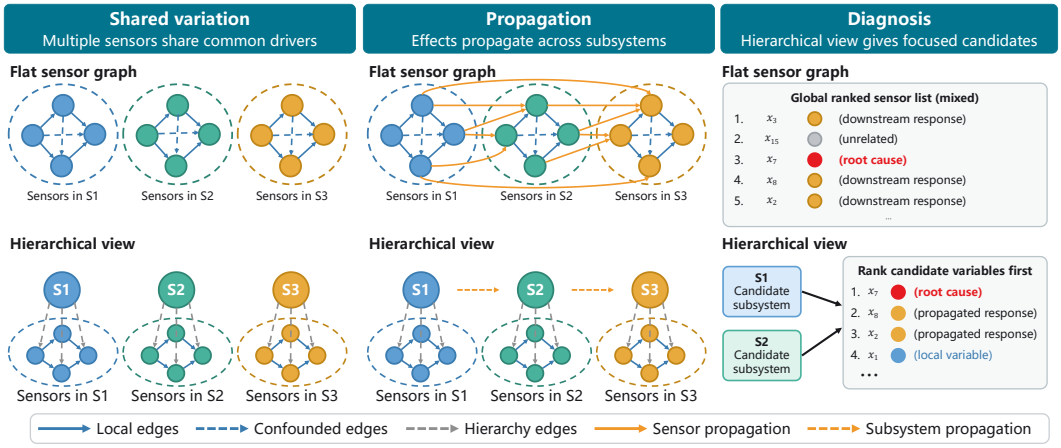

Figure 1: **Motivation: from flat to hierarchical Granger RCA.** The three columns illustrate shared variation, disturbance propagation, and root cause localization. A flat variable graph can mix shared subsystem variation with local dependencies and can represent cross-subsystem propagation with many variable-level edges. The hierarchical view separates subsystem-level propagation from local variable interactions and uses candidate subsystem screening to focus the final variable ranking.

and their temporal dependencies from normal-operation data. This setting raises two questions: can subsystem semantics help separate shared subsystem dynamics from local variable-level effects, and can this hierarchy improve root localization beyond a flat variable-level model?

We introduce HiRCA, a hierarchical Granger-causal framework for RCA in industrial multivariate time series. Sensor-specific encoders and soft memberships first construct latent subsystem states. A temporal graph over these states models cross-subsystem propagation. At the variable level, each target is predicted by three additive branches. The self-history branch models the target's own temporal dynamics. The cross-variable branch captures predictive information from other variables' histories. The subsystem-context branch uses a target-excluded latent representation to capture variation shared within the target subsystem. Separating this shared context allows the variable-level paths to focus on additional predictive contributions from individual variables. During diagnosis, calibrated innovation scores are first aggregated to identify candidate subsystems. Variables in the candidate union are then prioritized in the final ranking to localize root variables.

The main contributions are summarized as follows:

- We formulate hierarchical Granger-causal RCA for industrial multivariate time series, separating cross-subsystem dynamics from conditional variable-level interactions under engineering subsystem membership.
- We develop a learnable subsystem representation and a two-level predictive model. A target-excluded subsystem context accounts for shared subsystem variation, while sparse variable-level paths capture additional predictive information between variables.
- We propose a two-stage localization procedure that first screens multiple candidate subsystems and then prioritizes their variables in the final ranking, enabling multiple root variables across interacting subsystems.
- We evaluate HiRCA on synthetic and real-world datasets. The experiments cover subsystem and variable level graph recovery, root cause localization, and an analysis of the two-stage inference procedure.

2 Related Work

Existing RCA methods extract abnormal evidence in different ways. Statistical methods identify variables whose distributions change between normal and anomalous periods (Shan et al., 2019).

Causal-graph methods infer root variables from learned dependency structures or localized causal discovery (Ikram et al., 2022; Xin et al., 2023). In industrial systems, these methods have been instantiated with Granger-based predictive models, causal-graph tracing, and Gaussian-process causal discovery (Lv et al., 2025; Ji et al., 2025; Kong et al., 2026). Reconstruction-based methods instead use reconstruction residuals from patterns learned during normal operation to detect and diagnose anomalies (Zhang et al., 2019). AERCA (Han et al., 2025) takes a predictive view by modeling anomalies as interventions on exogenous variables and scoring deviations from their normal distributions. Other work combines temporal knowledge mining with RCA to capture richer causal rules in multivariate time series (Wang et al., 2026). Despite these differences, most existing methods reason over variables at a single resolution. Industrial variables, however, are often organized into interacting subsystems. This structure suggests that root localization may benefit from modeling both subsystem-level propagation and variable-level interactions.

Hierarchical structure has been explored from several perspectives. Multi-level diagnosis methods model anomaly evolution or temporal causal relations across different levels (Wang et al., 2025; Pan et al., 2026). REASON learns causal relations within and across entity levels in interdependent systems and uses them to model fault propagation for root localization (Wang et al., 2023). Wang et al. (Wang et al., 2024a) organize process variables into layers using redundancy information and apply Granger analysis within and across these layers to trace disturbance propagation. Causal representation learning takes a different view and aims to recover latent causal variables and their dependencies from high-dimensional observations (Schölkopf et al., 2021). Temporal approaches learn latent causal processes from sequential data (Yao et al., 2021), while SPACY extends latent causal discovery to high-dimensional spatiotemporal observations (Wang et al., 2024b). MOSAIC further links identifiable latent factors to sparse groups of observed variables to obtain interpretable modules (Fan et al., 2026). Unlike redundancy-derived variable layers or predefined entity-level networks, our formulation learns latent subsystem states anchored by a soft engineering prior and conditions variable-level effects on a target-excluded subsystem context.

3 METHOD

3.1 PROBLEM SETUP AND HIERARCHICAL VIEW

Let $\{\mathbf{x}_t\}_{t=1}^{T}$, with $\mathbf{x}_t \in \mathbb{R}^d$, denote an industrial multivariate time series from K interacting subsystems. Given an anomalous event, our goal is to rank its root variables and localize their anomalous time windows. We assume an exclusive engineering reference membership: each observed variable belongs to exactly one subsystem according to an engineering membership map $b : [d] \to [K]$. Learned memberships relax this discrete reference assignment during optimization. The temporal subsystem graph and the local variable graph may have multiple incoming edges.

We model normal operation through two coupled resolutions,

$$z_{k,t} = F_k(\{z_{m,t-\ell}\}_{m,\ell}) + \varepsilon^z_{k,t}, \tag{1}$$

$$x_{i,t} = g_i(z_{b(i),t}, x_{i,t-1:t-p_x}, \{x_{j,t-\ell} : b(j) = b(i), j \neq i\}) + \varepsilon^x_{i,t}. \tag{2}$$

Here $\varepsilon^z_{k,t}$ and $\varepsilon^x_{i,t}$ denote exogenous innovations. Cross-subsystem dependence is modeled through the latent dynamics, while the variable-level mechanism captures additional within-subsystem dependencies.

Under stable normal mechanisms, propagated responses can be explained by their predictive inputs, while unexplained deviations appear in the target innovation. We therefore score innovations by how unlikely they are under normal operation. Appendix A states the corresponding predictive-support argument.

Figure 2 connects the representation modules, predictive branches, and diagnostic procedure. The upper graph G^z models latent temporal dependence across subsystems. The lower graph G^x models conditional variable dependencies given self-history and a target-excluded subsystem context.

3.2 HIERARCHY-INFORMED SUBSYSTEM REPRESENTATION

We allocate $K_{\max} > K$ subsystem slots. The K engineering subsystems initialize the reference structure, while the additional slots provide learnable capacity. For variable i, $\mathbf{B}_i \in \mathbb{R}^{K_{\max}}$ is a

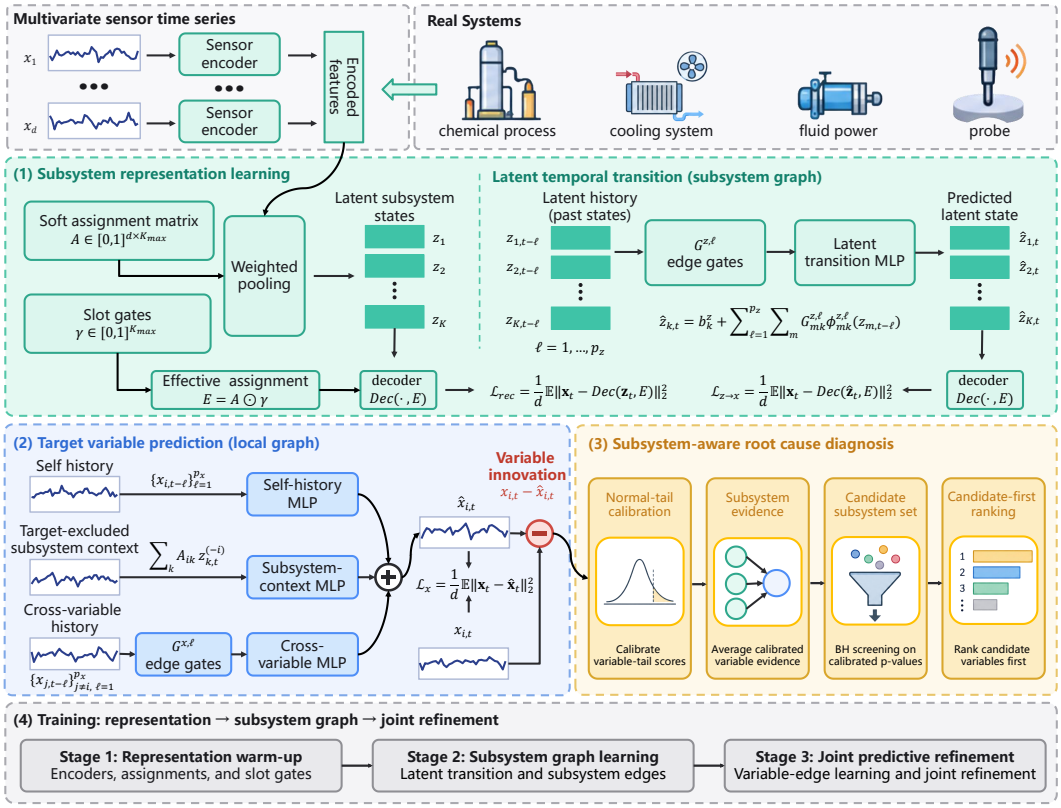

Figure 2: **Overview of HiRCA.** (1) Sensor-specific encoders and soft assignments construct latent subsystem states, while slot gates form the effective assignment for downstream modeling; the latent temporal graph captures cross-subsystem dynamics. (2) Each target variable is predicted from its self-history, a target-excluded subsystem context, and gated histories of other variables. (3) Calibrated variable innovations are aggregated for subsystem screening, followed by candidate-guided variable ranking. (4) Training proceeds through representation warm-up, subsystem graph learning, and joint predictive refinement.

learnable vector of membership logits. The soft assignment A_{ik}, activity gate γ_k, and gated assignment weight E_{ik} are

$$A_{ik} = \mathrm{softmax}(\mathbf{B}_i/\tau_A)_k, \qquad \gamma_k = \sigma(s_k), \qquad E_{ik} = A_{ik}\gamma_k. \tag{3}$$

The scalar s_k controls slot activity, and τ_A controls assignment concentration. Each sensor-specific encoder maps $x_{i,t}$ to a Gaussian mean $h_{i,t}$ and variance $v_{i,t}$. Because γ_k is shared by all variables assigned to slot k, it cancels under normalized pooling. We therefore define the subsystem posterior directly from the soft assignments:

$$\mu^z_{k,t} = \frac{\sum_j A_{jk} h_{j,t}}{\sum_j A_{jk}}, \qquad v^z_{k,t} = \frac{\sum_j A^2_{jk} v_{j,t}}{\left(\sum_j A_{jk}\right)^2}. \tag{4}$$

The activity gate γ_k instead controls gated downstream uses through the effective assignment, including decoder weighting, co-membership support, and active-slot selection. The normalized latent state, latent transition, and subsystem-context mixture remain ungated, so the slot gate acts as a structural/selection mechanism rather than rescaling latent dynamics. Training samples $z_{k,t} = \mu^z_{k,t} + \sqrt{v^z_{k,t}}\xi$, $\xi \sim \mathcal{N}(0,1)$, for reconstruction; latent transitions use posterior means throughout, and inference uses means for all latent states.

For prediction of variable i, the subsystem context excludes its current encoded observation:

$$z_{k,t}^{(-i)} = \frac{\sum_j A_{jk} h_{j,t} - A_{ik} h_{i,t}}{\sum_j A_{jk} - A_{ik}}. \tag{5}$$

This target-excluded context captures contemporaneous variation shared by other variables in the same subsystem, while lagged variable histories capture additional directed predictive dependencies. Appendix D specifies the treatment of slots with negligible remaining peer mass.

3.3 Hierarchical Granger model

Cross-subsystem dynamics The normal-regime latent predictor is

$$\widehat{z}_{k,t} = b_k^z + \sum_{\ell=1}^{p_z} \sum_{m=1}^{K_{\max}} G_{mk}^{z,\ell} \phi_{mk}^{z,\ell}(\mu_{m,t-\ell}^z). \tag{6}$$

Here b_k^z is an intercept, $\phi_{mk}^{z,\ell}$ is an edge-specific residual MLP, and $G_{mk}^{z,\ell}$ is a binary edge variable parameterized by a learned inclusion probability $\pi_{mk}^{z,\ell}$. The edge function models the predictive contribution of its source, whereas $G_{mk}^{z,\ell}$ determines whether the lag-ℓ transition from subsystem m to subsystem k is included. We optimize these binary variables through differentiable relaxations (Maddison et al., 2016; Jang et al., 2016). For structural evaluation, $\pi_{mk}^{z,\ell} = \Pr(G_{mk}^{z,\ell} = 1 \mid \mathcal{D}_0)$ serves as the subsystem-edge score.

Within-subsystem variable mechanisms Let $\bar{E}$ be the row-normalized effective assignment. The soft co-membership support is $C_{ji} = \sum_k \bar{E}_{jk} \bar{E}_{ik}$ for $j \neq i$, with $C_{ii} = 0$. Each target has three additive branches:

$$\begin{aligned} \widehat{x}_{i,t} = f_i^{\text{self}}(x_{i,t-1:t-p_x}) + f_i^{\text{sub}}\left(\sum_k A_{ik} z_{k,t}^{(-i)}\right) \\ + \sum_{\ell=1}^{p_x} \sum_{j \neq i} C_{ji} G_{ji}^{x,\ell} \phi_{ji}^{x,\ell}(x_{j,t-\ell}). \end{aligned} \tag{7}$$

The self branch models the target's own temporal dynamics. The subsystem branch captures contemporaneous variation shared with other variables in the same subsystem, while excluding the target itself. The cross-variable branch captures additional predictive information from other variables' lagged histories. The self and subsystem branches use sensor-specific MLPs, while the cross branch sums edge-specific residual MLP outputs weighted by co-membership and graph gates.

Each lag-specific variable edge is parameterized by a low-rank source–target score. During training, Gumbel-softmax assignments and relaxed graph gates enable differentiable optimization. At evaluation, prediction retains deterministic soft assignments and gate probabilities. Active slots $\mathcal{K}_{\text{act}}$ are selected adaptively using preliminary membership counts and effective assignment mass. For discrete graph support, $\widehat{b}_E(i) = \arg\max_{k \in \mathcal{K}_{\text{act}}} E_{ik}$, whereas subsystem screening uses $\widehat{b}_A(i) = \arg\max_{k \in \mathcal{K}_{\text{act}}} A_{ik}$.

Graph recovery scores The subsystem-edge score is $S_{mk}^{z,\ell} = \pi_{mk}^{z,\ell}$. We score variable edges by the norm of their input-to-output path weights, restrict them to the hard learned subsystem support, and normalize within each target. Appendix B gives the expression for $S_{ji}^{x,\ell}$.

3.4 Training objective and staged optimization

The training objective combines variable prediction, observation reconstruction, and decoded latent prediction:

$$\mathcal{L} = \lambda_x \mathcal{L}_x + \lambda_{\text{rec}} \mathcal{L}_{\text{rec}} + \lambda_{z \to x} \mathcal{L}_{z \to x} + \Omega. \tag{8}$$

A central component of the hierarchy constraint is the engineering reference partition. Let

$$R_{ij}^{\text{ref}} = \mathbb{I}[b(i) = b(j)]. \tag{9}$$

Table 1: Subsystem summary-graph recovery.

Model	Linear				Nonlinear			
	F1↑	AUC-PR↑	AUC-ROC↑	HD↓	F1↑	AUC-PR↑	AUC-ROC↑	HD↓
MOSAIC	$0.261_{\pm 0.153}$	$0.514_{\pm 0.115}$	$0.581_{\pm 0.087}$	$0.417_{\pm 0.059}$	$0.327_{\pm 0.069}$	$0.443_{\pm 0.077}$	$0.512_{\pm 0.116}$	$0.400_{\pm 0.037}$
SPACY	$0.133_{\pm 0.183}$	$0.350_{\pm 0.023}$	$0.525_{\pm 0.034}$	$0.333_{\pm 0.000}$	$0.147_{\pm 0.202}$	$0.342_{\pm 0.019}$	$0.487_{\pm 0.068}$	$0.400_{\pm 0.091}$
HiRCA	$0.760_{\pm 0.329}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.100_{\pm 0.137}$	$0.760_{\pm 0.329}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.100_{\pm 0.137}$

We regularize the learned pairwise co-membership by

$$\mathcal{R}_{\text{ref}} = \frac{1}{d(d-1)} \sum_{i \neq j} \left[(AA^\top)_{ij} - R^{\text{ref}}_{ij} \right]^2 . \tag{10}$$

This term encourages the learned soft memberships to preserve known engineering subsystem relations while retaining learnable assignment flexibility.

The remaining terms in Ω regularize the latent posterior, slot structure, graph sparsity, and predictive paths; their definitions are given in Appendix C.

Training has three stages: representation warm-up, subsystem graph learning, and joint predictive refinement. Each stage uses normal-operation windows, with normal validation loss for model selection. Appendix C lists all objective coefficients; the reported experiments use $\lambda_x = 1$, $\lambda_{\text{rec}} = 0.5$, and $\lambda_{z \to x} = 0.5$.

3.5 Two-stage root cause inference

For a diagnosis window n with valid prediction times $\mathcal{T}_n$, let $e_{i,t} = x_{i,t} - \widehat{x}_{i,t}$ be the variable innovation. We estimate a robust center m_i and scale d_i from normal-operation innovations and standardize the residual as $\widetilde{e}_{i,t} = (e_{i,t} - m_i)/d_i$. We average $|\widetilde{e}_{i,t}|$ over the window to obtain its magnitude $a_{i,n}$ (Appendix D). Let $\overline{F}_i^{(0)}$ denote the upper-tail survival function of the corresponding magnitudes under normal operation. Variable evidence is the calibrated normal-tail surprise

$$s_{i,n} = -\log_{10} \overline{F}_i^{(0)}(a_{i,n}). \tag{11}$$

This transformation places variables with different residual scales on a comparable evidence scale: a large $s_{i,n}$ indicates that the observed prediction error is rare relative to that variable's normal behavior. Calibration details and numerical safeguards are given in Appendix D.

Stage 1: subsystem screening Using the screening membership defined above, let $V_k = \{i : \widehat{b}_A(i) = k\}$ contain the variables assigned to active slot k. We aggregate member evidence by

$$u_{k,n} = \frac{1}{|V_k|} \sum_{i \in V_k} s_{i,n}, \qquad p_{k,n} = \frac{1 + \sum_{m \in \mathcal{N}_0} \mathbb{I}[u_{k,m} \geq u_{k,n}]}{1 + |\mathcal{N}_0|}, \tag{12}$$

where $\mathcal{N}_0$ contains normal calibration windows. The mean normalizes for subsystem size while aggregating consistent abnormal evidence across its members. We then apply the Benjamini–Hochberg procedure to $p_{k,n}$ to obtain a candidate subsystem set $\widehat{\mathcal{R}}_n$ (Benjamini & Hochberg, 1995). The significance level and empty-set safeguard are given in Appendix D.

Stage 2: variable localization The selected subsystems define $\widehat{\mathcal{V}}_n = \bigcup_{k \in \widehat{\mathcal{R}}_n} V_k$. Variables in this union are ranked first by $s_{i,n}$, followed by the remaining variables, with descending evidence order within both groups. Time-variable localization applies the same candidate priority and evidence ordering across an event's windows, with each time index identifying a window.

4 Experiments

We evaluate HiRCA on two tasks: hierarchical graph recovery and root cause localization. We also analyze the effect of the two-stage inference procedure by comparing it with direct ranking. Tables

Table 2: Within-subsystem variable summary-graph recovery.

Model	Linear				Nonlinear			
	F1↑	AUC-PR↑	AUC-ROC↑	HD↓	F1↑	AUC-PR↑	AUC-ROC↑	HD↓
AERCA	$0.239_{\pm 0.144}$	$0.357_{\pm 0.088}$	$0.594_{\pm 0.087}$	$0.260_{\pm 0.045}$	$0.287_{\pm 0.117}$	$0.346_{\pm 0.088}$	$0.588_{\pm 0.083}$	$0.305_{\pm 0.104}$
CUTS	$0.298_{\pm 0.065}$	$0.495_{\pm 0.076}$	$0.474_{\pm 0.091}$	$0.470_{\pm 0.072}$	$0.290_{\pm 0.068}$	$0.495_{\pm 0.091}$	$0.471_{\pm 0.093}$	$0.510_{\pm 0.042}$
GVAR	$0.162_{\pm 0.098}$	$0.274_{\pm 0.043}$	$0.493_{\pm 0.049}$	$0.315_{\pm 0.034}$	$0.115_{\pm 0.082}$	$0.273_{\pm 0.021}$	$0.473_{\pm 0.051}$	$0.300_{\pm 0.059}$
PCMCI	$0.386_{\pm 0.076}$	$0.566_{\pm 0.070}$	$0.586_{\pm 0.092}$	$0.195_{\pm 0.011}$	$0.386_{\pm 0.076}$	$0.563_{\pm 0.073}$	$0.584_{\pm 0.108}$	$0.195_{\pm 0.011}$
PCMCI+	$0.219_{\pm 0.131}$	$0.361_{\pm 0.103}$	$0.453_{\pm 0.104}$	$0.240_{\pm 0.029}$	$0.260_{\pm 0.112}$	$0.377_{\pm 0.089}$	$0.450_{\pm 0.088}$	$0.235_{\pm 0.022}$
TCDF	$0.231_{\pm 0.058}$	$0.335_{\pm 0.044}$	$0.531_{\pm 0.059}$	$0.290_{\pm 0.045}$	$0.176_{\pm 0.065}$	$0.327_{\pm 0.046}$	$0.528_{\pm 0.060}$	$0.295_{\pm 0.021}$
VAR	$0.505_{\pm 0.089}$	$0.671_{\pm 0.085}$	$0.725_{\pm 0.094}$	$0.343_{\pm 0.063}$	$0.501_{\pm 0.071}$	$0.663_{\pm 0.087}$	$0.716_{\pm 0.096}$	$0.345_{\pm 0.065}$
cLSTM	$0.400_{\pm 0.000}$	$0.250_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.750_{\pm 0.000}$	$0.400_{\pm 0.000}$	$0.250_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.750_{\pm 0.000}$
cMLP	$0.100_{\pm 0.105}$	$0.265_{\pm 0.026}$	$0.501_{\pm 0.036}$	$0.320_{\pm 0.027}$	$0.136_{\pm 0.103}$	$0.270_{\pm 0.021}$	$0.518_{\pm 0.035}$	$0.320_{\pm 0.062}$
eSRU	$0.300_{\pm 0.126}$	$0.421_{\pm 0.064}$	$0.528_{\pm 0.085}$	$0.215_{\pm 0.029}$	$0.300_{\pm 0.126}$	$0.427_{\pm 0.067}$	$0.532_{\pm 0.088}$	$0.215_{\pm 0.029}$
HiRCA	$0.721_{\pm 0.081}$	$0.684_{\pm 0.091}$	$0.871_{\pm 0.061}$	$0.145_{\pm 0.041}$	$0.678_{\pm 0.091}$	$0.690_{\pm 0.074}$	$0.885_{\pm 0.051}$	$0.205_{\pm 0.086}$

report means and sample standard deviations across five seeds. Red, blue, and yellow indicate the best, second-best, and third-best mean results for each dataset and metric.

4.1 Data and evaluation protocol

Synthetic systems Linear and Nonlinear each contain $d = 20$ observed variables and $K = 4$ subsystems. The data are generated hierarchically. Let $a_{mk}^{z,\ell}$ denote a sampled latent temporal edge coefficient, α_i the single nonzero loading from the parent subsystem $b(i)$ to variable i, and $\mathrm{pa}(i)$ a sampled within-subsystem parent of i. A compact form of the generator is

$$z_{k,t} = \sum_{m=1}^{K} \sum_{\ell} a_{mk}^{z,\ell} \phi_{mk}^{z}(z_{m,t-\ell}) + \varepsilon_{k,t}^{z},$$
$$x_{i,t} = \phi_i^{\mathrm{obs}}(\alpha_i z_{b(i),t}) + \phi_i^{\mathrm{self}}(x_{i,t-1:t-p_x}) + \phi_i^{\mathrm{peer}}(x_{\mathrm{pa}(i),t-1:t-p_x}) + \varepsilon_{i,t}^{x}. \tag{13}$$

Linear uses identity latent and observation maps with linear self-history and peer responses. Nonlinear uses $\tanh$ for latent self-effects and observation mapping, $\sin$ for cross-subsystem effects, and weighted $\tanh$ responses for self-history and peers. Test events jointly perturb the innovations of selected observed variables and their parent subsystems, with signed amplitudes a and $0.65a$, respectively. Both systems use variable lag $p_x = 1$, latent lag $p_z = 1$, and fault strength $a = 3$; root labels identify directly perturbed observed variables. Each event spans one to four selected subsystems. Each seed contains ten normal training trajectories, two normal validation trajectories, and 100 test trajectories; each trajectory has 500 time points. Windows have length eight and stride eight, and normalization statistics are estimated from normal training data.

Graph recovery Subsystem recovery compares HiRCA with MOSAIC (Fan et al., 2026) and SPACY (Wang et al., 2024b). Variable-level recovery compares against AERCA (Han et al., 2025), CUTS (Cheng et al., 2023), GVAR (Marcinkevičs & Vogt, 2021), PCMCI (Runge et al., 2019), PCMCI+ (Runge, 2020), TCDF (Nauta et al., 2019), VAR (Lütkepohl et al., 2005), cLSTM/cMLP (Tank et al., 2021), and eSRU (Khanna & Tan, 2019). Following AERCA (Han et al., 2025), we report F1, normalized Hamming distance (HD), AUC-PR, and AUC-ROC. Lagged edges are collapsed by their maximum over lag, and self-loops are excluded. Variable-level metrics are micro-aggregated over within-subsystem candidate edges. For methods that output continuous edge scores, we use the same score-based thresholding rule at each graph level; definitions and calibration are given in Appendices E and D.

Root localization We follow AERCA's ranking metrics (Han et al., 2025): AC@1, 3, 5, 10 and Avg@10 for variable ranking, and AC*@1, 10, 100, 500 and Avg*@500 for joint time-variable ranking. Appendix E gives the metric definitions and event-level aggregation. RCA baselines are ϵ-Diagnosis (Shan et al., 2019), RCD (Ikram et al., 2022), AERCA (Han et al., 2025), and CausalRCA (Xin et al., 2023). CausalRCA provides a single event-level variable ranking rather than the time-resolved variable scores required for joint time-variable localization. Its AC* metrics are

Table 3: Synthetic RCA: variable and time-variable ranking.

Dataset	Model	Variable ranking					Time-variable ranking				
		AC@1↑	AC@3↑	AC@5↑	AC@10↑	Avg@10↑	AC*@1↑	AC*@10↑	AC*@100↑	AC*@500↑	Avg*@500↑
Linear	ϵ-Diagnosis	$0.460_{\pm 0.028}$	$0.495_{\pm 0.045}$	$0.508_{\pm 0.044}$	$0.602_{\pm 0.030}$	$0.522_{\pm 0.036}$	$0.006_{\pm 0.005}$	$0.011_{\pm 0.005}$	$0.077_{\pm 0.018}$	$0.406_{\pm 0.033}$	$0.201_{\pm 0.023}$
	RCD	$0.688_{\pm 0.054}$	$0.665_{\pm 0.035}$	$0.653_{\pm 0.014}$	$0.680_{\pm 0.025}$	$0.661_{\pm 0.024}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.142_{\pm 0.019}$	$0.491_{\pm 0.033}$	$0.290_{\pm 0.024}$
	AERCA	$0.934_{\pm 0.019}$	$0.787_{\pm 0.027}$	$0.696_{\pm 0.024}$	$0.598_{\pm 0.022}$	$0.711_{\pm 0.021}$	$0.932_{\pm 0.019}$	$0.541_{\pm 0.025}$	$0.487_{\pm 0.016}$	$0.494_{\pm 0.014}$	$0.494_{\pm 0.016}$
	CausalRCA	$0.414_{\pm 0.064}$	$0.455_{\pm 0.025}$	$0.493_{\pm 0.029}$	$0.614_{\pm 0.031}$	$0.504_{\pm 0.019}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.878_{\pm 0.053}$	$0.887_{\pm 0.035}$	$0.914_{\pm 0.033}$	$0.942_{\pm 0.010}$	$0.907_{\pm 0.027}$	$0.572_{\pm 0.112}$	$0.877_{\pm 0.022}$	$0.982_{\pm 0.010}$	$1.000_{\pm 0.000}$	$0.986_{\pm 0.005}$
Nonlinear	ϵ-Diagnosis	$0.444_{\pm 0.076}$	$0.487_{\pm 0.055}$	$0.513_{\pm 0.048}$	$0.598_{\pm 0.030}$	$0.522_{\pm 0.044}$	$0.006_{\pm 0.005}$	$0.011_{\pm 0.004}$	$0.076_{\pm 0.013}$	$0.408_{\pm 0.033}$	$0.205_{\pm 0.023}$
	RCD	$0.996_{\pm 0.005}$	$0.914_{\pm 0.015}$	$0.827_{\pm 0.013}$	$0.756_{\pm 0.014}$	$0.836_{\pm 0.009}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.201_{\pm 0.013}$	$0.572_{\pm 0.026}$	$0.370_{\pm 0.016}$
	AERCA	$0.980_{\pm 0.012}$	$0.901_{\pm 0.014}$	$0.802_{\pm 0.025}$	$0.598_{\pm 0.018}$	$0.782_{\pm 0.016}$	$0.980_{\pm 0.012}$	$0.595_{\pm 0.018}$	$0.494_{\pm 0.014}$	$0.494_{\pm 0.014}$	$0.500_{\pm 0.013}$
	CausalRCA	$0.286_{\pm 0.041}$	$0.416_{\pm 0.018}$	$0.488_{\pm 0.031}$	$0.628_{\pm 0.034}$	$0.483_{\pm 0.024}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.910_{\pm 0.037}$	$0.902_{\pm 0.019}$	$0.929_{\pm 0.023}$	$0.952_{\pm 0.011}$	$0.923_{\pm 0.017}$	$0.588_{\pm 0.102}$	$0.905_{\pm 0.025}$	$0.996_{\pm 0.002}$	$1.000_{\pm 0.000}$	$0.993_{\pm 0.002}$

therefore reported as N/A. Real datasets comprise the Tennessee Eastman Process (TEP) (Downs & Vogel, 1993) and Coolant, Hydraulics, and Probe from the causRCA manufacturing benchmark (Mehling et al., 2026).

4.2 Hierarchical Graph Recovery

HiRCA achieved the highest subsystem-graph F1 of 0.760 on both synthetic datasets, with AUC-PR and AUC-ROC of 1.000 (Table 1). At the variable level, it achieved the highest F1, AUC-PR, and AUC-ROC on both datasets (Table 2). Variable-graph F1 reached 0.721 on Linear and 0.678 on Nonlinear. These results show strong edge ranking at the subsystem level and accurate recovery of fine-grained predictive dependencies.

4.3 Root Cause Localization

We evaluate root cause localization at two granularities. Tables 3 and 4 jointly report variable ranking and time-variable ranking results for the synthetic and real datasets, respectively.

Variable ranking On both synthetic datasets, HiRCA achieves the highest Avg@10, with 0.907 on Linear and 0.923 on Nonlinear. Its rankings place root variables near the top of the candidate list across the two dynamics settings. On the real datasets, HiRCA achieves the highest Avg@10 on TEP, with 0.472. It obtains the second-highest Avg@10 on Coolant and Probe and the third-highest on Hydraulics. These results indicate competitive variable ranking performance across datasets with different dimensions and fault characteristics.

Time-variable ranking On the synthetic benchmarks, HiRCA achieves the highest Avg*@500 among the compared methods, reaching 0.986 on Linear and 0.993 on Nonlinear. On real data, it leads on Coolant, Hydraulics, and Probe, with 0.577, 0.303, and 0.640, respectively. On TEP, its Avg*@500 is 0.127, compared with 0.132 for AERCA. These results complement the variable ranking evaluation: AC measures whether root variables are ranked highly, whereas AC* jointly evaluates the ranking of variables and their temporal locations.

4.4 Analysis of the Two-Stage Inference Procedure

We compare the two-stage inference procedure with direct ranking using the same trained hierarchical predictor and the same calibrated innovation scores. The only difference is whether candidate-subsystem screening is applied before the final variable ordering. Table 5 reports the corresponding variable-ranking and time-variable-ranking results on the synthetic datasets.

Variable ranking Two-stage inference achieves Avg@10 scores of 0.907 on Linear and 0.923 on Nonlinear, compared with 0.903 and 0.922 for direct ranking. The consistent gains indicate that candidate-subsystem screening improves average variable ranking on both synthetic settings.

Time-variable ranking For joint time-variable ranking, two-stage inference achieves Avg*@500 scores of 0.986 on Linear and 0.993 on Nonlinear, compared with 0.977 and 0.992 for direct rank-

Table 4: Real-data RCA: variable and time-variable ranking.

Dataset	Model	Variable ranking					Time-variable ranking				
		AC@1↑	AC@3↑	AC@5↑	AC@10↑	Avg@10↑	AC*@1↑	AC*@10↑	AC*@100↑	AC*@500↑	Avg*@500↑
Coolant	ϵ-Diagnosis	$0.421_{\pm 0.001}$	$0.409_{\pm 0.001}$	$0.677_{\pm 0.001}$	$1.000_{\pm 0.000}$	$0.723_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$
	RCD	$0.605_{\pm 0.000}$	$0.446_{\pm 0.002}$	$0.640_{\pm 0.003}$	$1.000_{\pm 0.000}$	$0.736_{\pm 0.002}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.020_{\pm 0.000}$	$0.378_{\pm 0.002}$	$0.224_{\pm 0.001}$
	AERCA	$0.291_{\pm 0.058}$	$0.229_{\pm 0.074}$	$0.363_{\pm 0.048}$	$1.000_{\pm 0.000}$	$0.514_{\pm 0.032}$	$0.280_{\pm 0.000}$	$0.292_{\pm 0.033}$	$0.113_{\pm 0.046}$	$0.182_{\pm 0.107}$	$0.154_{\pm 0.064}$
	CausalRCA	$0.648_{\pm 0.125}$	$0.793_{\pm 0.058}$	$0.915_{\pm 0.044}$	$1.000_{\pm 0.000}$	$0.891_{\pm 0.031}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.721_{\pm 0.011}$	$0.650_{\pm 0.006}$	$0.785_{\pm 0.005}$	$1.000_{\pm 0.000}$	$0.839_{\pm 0.002}$	$0.792_{\pm 0.140}$	$0.871_{\pm 0.089}$	$0.695_{\pm 0.026}$	$0.536_{\pm 0.005}$	$0.577_{\pm 0.015}$
Hydraulics	ϵ-Diagnosis	$0.136_{\pm 0.000}$	$0.409_{\pm 0.000}$	$0.773_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.682_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$
	RCD	$0.209_{\pm 0.004}$	$0.225_{\pm 0.003}$	$0.790_{\pm 0.001}$	$1.000_{\pm 0.000}$	$0.648_{\pm 0.001}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.005_{\pm 0.000}$	$0.332_{\pm 0.003}$	$0.155_{\pm 0.001}$
	AERCA	$0.270_{\pm 0.084}$	$0.269_{\pm 0.158}$	$0.363_{\pm 0.136}$	$1.000_{\pm 0.000}$	$0.541_{\pm 0.088}$	$0.009_{\pm 0.020}$	$0.024_{\pm 0.053}$	$0.021_{\pm 0.014}$	$0.312_{\pm 0.129}$	$0.180_{\pm 0.086}$
	CausalRCA	$0.245_{\pm 0.123}$	$0.632_{\pm 0.083}$	$0.814_{\pm 0.071}$	$1.000_{\pm 0.000}$	$0.750_{\pm 0.039}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.172_{\pm 0.000}$	$0.339_{\pm 0.000}$	$0.838_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.680_{\pm 0.000}$	$0.182_{\pm 0.000}$	$0.200_{\pm 0.000}$	$0.330_{\pm 0.000}$	$0.291_{\pm 0.000}$	$0.303_{\pm 0.000}$
Probe	ϵ-Diagnosis	$0.575_{\pm 0.002}$	$0.888_{\pm 0.002}$	$0.958_{\pm 0.002}$	$1.000_{\pm 0.000}$	$0.914_{\pm 0.001}$	$0.059_{\pm 0.036}$	$0.034_{\pm 0.007}$	$0.043_{\pm 0.002}$	$0.088_{\pm 0.002}$	$0.072_{\pm 0.002}$
	RCD	$0.268_{\pm 0.005}$	$0.738_{\pm 0.005}$	$0.996_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.832_{\pm 0.002}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.036_{\pm 0.001}$	$0.692_{\pm 0.005}$	$0.266_{\pm 0.004}$
	AERCA	$0.237_{\pm 0.047}$	$0.348_{\pm 0.017}$	$0.535_{\pm 0.038}$	$1.000_{\pm 0.000}$	$0.619_{\pm 0.013}$	$0.000_{\pm 0.000}$	$0.101_{\pm 0.083}$	$0.171_{\pm 0.063}$	$0.396_{\pm 0.047}$	$0.246_{\pm 0.030}$
	CausalRCA	$0.159_{\pm 0.053}$	$0.344_{\pm 0.075}$	$0.612_{\pm 0.053}$	$1.000_{\pm 0.000}$	$0.633_{\pm 0.032}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.683_{\pm 0.023}$	$0.751_{\pm 0.007}$	$0.876_{\pm 0.061}$	$1.000_{\pm 0.000}$	$0.877_{\pm 0.023}$	$0.353_{\pm 0.487}$	$0.792_{\pm 0.101}$	$0.589_{\pm 0.003}$	$0.686_{\pm 0.025}$	$0.640_{\pm 0.014}$
TEP	ϵ-Diagnosis	$0.026_{\pm 0.002}$	$0.056_{\pm 0.002}$	$0.092_{\pm 0.003}$	$0.183_{\pm 0.005}$	$0.102_{\pm 0.003}$	$0.027_{\pm 0.037}$	$0.021_{\pm 0.009}$	$0.020_{\pm 0.004}$	$0.021_{\pm 0.002}$	$0.020_{\pm 0.003}$
	RCD	$0.144_{\pm 0.002}$	$0.166_{\pm 0.002}$	$0.331_{\pm 0.001}$	$0.622_{\pm 0.001}$	$0.378_{\pm 0.001}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.000_{\pm 0.000}$	$0.091_{\pm 0.002}$	$0.043_{\pm 0.002}$
	AERCA	$0.151_{\pm 0.003}$	$0.139_{\pm 0.005}$	$0.181_{\pm 0.009}$	$0.288_{\pm 0.019}$	$0.197_{\pm 0.010}$	$0.133_{\pm 0.000}$	$0.133_{\pm 0.000}$	$0.131_{\pm 0.001}$	$0.139_{\pm 0.001}$	$0.132_{\pm 0.001}$
	CausalRCA	$0.013_{\pm 0.030}$	$0.120_{\pm 0.051}$	$0.200_{\pm 0.047}$	$0.300_{\pm 0.085}$	$0.191_{\pm 0.040}$	N/A	N/A	N/A	N/A	N/A
	HiRCA	$0.231_{\pm 0.018}$	$0.423_{\pm 0.017}$	$0.509_{\pm 0.015}$	$0.564_{\pm 0.011}$	$0.472_{\pm 0.010}$	$0.000_{\pm 0.000}$	$0.007_{\pm 0.009}$	$0.132_{\pm 0.006}$	$0.141_{\pm 0.004}$	$0.127_{\pm 0.005}$

Table 5: Synthetic comparison of two-stage inference and direct ranking.

Dataset	Model	Variable ranking					Time-variable ranking				
		AC@1↑	AC@3↑	AC@5↑	AC@10↑	Avg@10↑	AC*@1↑	AC*@10↑	AC*@100↑	AC*@500↑	Avg*@500↑
Linear	Direct ranking	$0.878_{\pm 0.053}$	$0.885_{\pm 0.035}$	$0.903_{\pm 0.029}$	$0.935_{\pm 0.011}$	$0.903_{\pm 0.025}$	$0.572_{\pm 0.112}$	$0.876_{\pm 0.021}$	$0.977_{\pm 0.013}$	$0.990_{\pm 0.006}$	$0.977_{\pm 0.010}$
	HiRCA	$0.878_{\pm 0.053}$	$0.887_{\pm 0.035}$	$0.914_{\pm 0.033}$	$0.942_{\pm 0.010}$	$0.907_{\pm 0.027}$	$0.572_{\pm 0.112}$	$0.877_{\pm 0.022}$	$0.982_{\pm 0.010}$	$1.000_{\pm 0.000}$	$0.986_{\pm 0.005}$
Nonlinear	Direct ranking	$0.910_{\pm 0.037}$	$0.902_{\pm 0.019}$	$0.925_{\pm 0.024}$	$0.951_{\pm 0.010}$	$0.922_{\pm 0.017}$	$0.588_{\pm 0.102}$	$0.905_{\pm 0.025}$	$0.994_{\pm 0.003}$	$1.000_{\pm 0.001}$	$0.992_{\pm 0.002}$
	HiRCA	$0.910_{\pm 0.037}$	$0.902_{\pm 0.019}$	$0.929_{\pm 0.023}$	$0.952_{\pm 0.011}$	$0.923_{\pm 0.017}$	$0.588_{\pm 0.102}$	$0.905_{\pm 0.025}$	$0.996_{\pm 0.002}$	$1.000_{\pm 0.000}$	$0.993_{\pm 0.002}$

ing, respectively. These results show a improvement from candidate-subsystem screening on both synthetic datasets.

Real-data results The real-data comparison in Appendix F shows that the effect of two-stage inference is dataset dependent. For variable ranking, two-stage inference improves Avg@10 on Probe and TEP, matches direct ranking on Coolant, and is lower on Hydraulics. For time-variable ranking, the two approaches are identical or close on most datasets.

The different behavior on synthetic and real datasets is closely related to the quality of the learned subsystem partition. In the synthetic systems, the engineering partition is consistent with the data-generating process by construction, so subsystem-level abnormal evidence can reliably guide the second-stage variable ranking. As a result, the two-stage procedure yields consistent improvements over direct ranking. In real industrial datasets, however, the available engineering partition may be coarse, incomplete, or not fully aligned with the statistical dependencies reflected in the observed data. In this case, errors in subsystem assignment can weaken the candidate-subsystem screening step and may even suppress relevant variables outside the selected subsystems. This explains why the benefit of two-stage inference is less consistent on the real datasets.

5 Conclusion

This paper introduced HiRCA, a hierarchical Granger-causal framework for root cause analysis in industrial multivariate time series. HiRCA separates cross-subsystem propagation from local variable-level dependencies and combines this hierarchy with calibrated innovations for coarse-to-fine root localization. On the synthetic benchmarks, HiRCA shows strong hierarchical graph-recovery performance and achieves Avg@10 scores above 0.90 for root cause localization. Across four real-world datasets, HiRCA ranks among the top three in nearly all evaluations, including 16 top-two variable ranking results and 15 best time-variable ranking results. It also obtains the highest Avg*@500 on three of the four real datasets. These results demonstrate competitive root localization performance across subsystem, variable, and temporal resolutions.

AI USE STATEMENT

We used generative AI tools to assist with writing and language polishing, literature retrieval and discovery, research ideation and execution, and the drafting of parts of the paper. The tools were used to improve clarity and presentation, help identify potentially relevant literature, discuss research ideas and experimental choices, and assist with drafting and revising text. All AI-assisted content was reviewed and revised by the authors. Relevant literature and factual claims were checked against the original sources, and research ideas and technical suggestions were evaluated by the authors before being incorporated into the work. The authors made the final decisions on the methodology, experiments, analysis, and presentation. We take full responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

Han-Sheng Chen, Zhengbing Yan, Xuelei Zhang, Yi Liu, and Yuan Yao. Root cause diagnosis of process faults using conditional granger causality analysis and maximum spanning tree. *IFAC-PapersOnLine*, 51(18):381–386, 2018.

Yuxiao Cheng, Runzhao Yang, Tingxiong Xiao, Zongren Li, Jinli Suo, Kunlun He, and Qionghai Dai. Cuts: Neural causal discovery from irregular time-series data. *arXiv preprint arXiv:2302.07458*, 2023.

James J Downs and Ernest F Vogel. A plant-wide industrial process control problem. *Computers & chemical engineering*, 17(3):245–255, 1993.

Shicheng Fan, Nour Elhendawy, Jianle Sun, Ke Fang, Kun Zhang, Yihang Wang, and Lu Cheng. Mosaic: Module discovery via sparse additive identifiable causal learning for scientific time series. *arXiv preprint arXiv:2605.05524*, 2026.

Xiao Han, Saima Absar, Lu Zhang, and Shuhan Yuan. Root cause analysis of anomalies in multivariate time series through granger causal discovery. In *International Conference on Learning Representations*, volume 2025, pp. 45137–45153, 2025.

Azam Ikram, Sarthak Chakraborty, Subrata Mitra, Shiv Saini, Saurabh Bagchi, and Murat Kocaoglu. Root cause analysis of failures in microservices through causal discovery. *Advances in Neural Information Processing Systems*, 35:31158–31170, 2022.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Shuaihang Ji, Jinjiang Wang, Zheng Wang, and Fengli Zhang. An explainable causal graph-based method for diagnosis and root cause analysis in process systems. *Process Safety and Environmental Protection*, pp. 107953, 2025.

Saurabh Khanna and Vincent YF Tan. Economy statistical recurrent units for inferring nonlinear granger causality. *arXiv preprint arXiv:1911.09879*, 2019.

Hyunmin Kong, Jitae Shin, and Chunsso Ahn. Gaussian process based causal discovery for root cause analysis in multivariate time series. In *Proceedings of the 2026 18th International Conference on Computer Research and Development*, pp. 261–266, 2026.

Bo Li, Qiang Li, Tingfeng Du, Dong Liu, Qiang Yang, Tianxiang Chen, Jing Xiong, Bo Peng, Junxiao Ren, and Ji Zhao. Research, application, and challenges of causal inference in industrial fault diagnosis: A survey. *Engineering Applications of Artificial Intelligence*, 158:111376, 2025a.

Zijian Li, Yifan Shen, Kaitao Zheng, Ruichu Cai, Xiangchen Song, Mingming Gong, Guangyi Chen, and Kun Zhang. On the identification of temporal causal representation with instantaneous dependence. In *International Conference on Learning Representations*, volume 2025, pp. 35203–35242, 2025b.

Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through l_0 regularization. *arXiv preprint arXiv:1712.01312*, 2017.

Helmut Lütkepohl et al. *New introduction to multiple time series analysis*, volume 1. springer Berlin, 2005.

Feiya Lv, Borui Yang, Shujian Yu, Shengwu Zou, Xiaolin Wang, Jinsong Zhao, and Chenglin Wen. A unified model integrating granger causality-based causal discovery and fault diagnosis in chemical processes. *Computers & Chemical Engineering*, 196:109028, 2025.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Ričards Marcinkevičs and Julia E Vogt. Interpretable models for granger causality using self-explaining neural networks. *arXiv preprint arXiv:2101.07600*, 2021.

Carl Willy Mehling, Sven Pieper, Tobias Lüke, Julius Döbelt, and Steffen Ihlenfeldt. Enabling joint benchmarking of automated root cause analysis and causal discovery in manufacturing using the causrca dataset. *Procedia CIRP*, 139:114–120, 2026.

Meike Nauta, Doina Bucur, and Christin Seifert. Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, 1(1):19, 2019.

Zhiwei Pan, Yang Hu, Joo-Ho Choi, Dingyang Zhang, Zheyuan Zhou, Shuyou Zhang, and Yiming Zhang. Temporal causal discovery-enhanced hierarchical monitoring for root cause diagnosis of aero-engine faults. *Knowledge-Based Systems*, pp. 115921, 2026.

S Joe Qin. Survey on data-driven industrial process monitoring and diagnosis. *Annual reviews in control*, 36(2):220–234, 2012.

Jakob Runge. Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets. In *Conference on uncertainty in artificial intelligence*, pp. 1388–1397. Pmlr, 2020.

Jakob Runge, Peer Nowack, Marlene Kretschmer, Seth Flaxman, and Dino Sejdinovic. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science advances*, 5 (11):eaau4996, 2019.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Huasong Shan, Yuan Chen, Haifeng Liu, Yunpeng Zhang, Xiao Xiao, Xiaofeng He, Min Li, and Wei Ding. ?-diagnosis: Unsupervised and real-time diagnosis of small-window long-tail latency in large-scale microservice platforms. In *The World Wide Web Conference*, pp. 3215–3222, 2019.

Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B Fox. Neural granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4267–4279, 2021.

Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewen Yin, and Surya N Kavuri. A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & chemical engineering*, 27(3):293–311, 2003.

Dongjie Wang, Zhengzhang Chen, Jingchao Ni, Liang Tong, Zheng Wang, Yanjie Fu, and Haifeng Chen. Interdependent causal networks for root cause localization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5051–5060, 2023.

Jian-Guo Wang, Rui Chen, Xiang-Yun Ye, Zhong-Tao Xie, Yuan Yao, and Li-Lan Liu. A hierarchical granger causality analysis framework based on information of redundancy for root cause diagnosis of process disturbances. *Computers & Chemical Engineering*, 182:108589, 2024a.

Kun Wang, Sumanth Varambally, Duncan Watson-Parris, Yi-An Ma, and Rose Yu. Discovering latent causal graphs from spatiotemporal data. *arXiv preprint arXiv:2411.05331*, 2024b.

Shengbo Wang, Yu Guo, Shaohua Huang, Qunfen Qi, Sai Geng, and Weiwei Qian. A multi-level root cause analysis method for production anomalies in manufacturing workshops. *Journal of Manufacturing Systems*, 80:776–793, 2025.

Xiaoliang Wang, Faming Lu, MengChu Zhou, and Qingtian Zeng. A new knowledge mining and root cause analysis methodology for multivariate time series. *IEEE/CAA Journal of Automatica Sinica*, 13(5):1054–1067, 2026.

Ruyue Xin, Peng Chen, and Zhiming Zhao. Causalrca: Causal inference based precise fine-grained root cause localization for microservice applications. *Journal of Systems and Software*, 203: 111724, 2023.

Weiran Yao, Yuewen Sun, Alex Ho, Changyin Sun, and Kun Zhang. Learning temporally causal latent processes from general temporal data. *arXiv preprint arXiv:2110.05428*, 2021.

Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 1409–1416, 2019.

A Predictive Support of the Hierarchical Graphs

This section characterizes the population-level predictive meaning of the two graphs used in HiRCA.

A general predictive-risk result. Let Y be a square-integrable prediction target, let $\mathcal{H}$ denote its full predictive context, and let $\mathcal{H}_{-a}$ denote the reduced context obtained by removing a candidate source a. Define the optimal squared prediction risk as

$$R(\mathcal{H}) = \inf_f \mathbb{E}\big[(Y - f(\mathcal{H}))^2\big]\,. \tag{14}$$

Proposition 1 (Predictive risk gap)**.** *Assume that $\sigma(\mathcal{H}_{-a}) \subseteq \sigma(\mathcal{H})$. Then*

$$\Delta_a = R(\mathcal{H}_{-a}) - R(\mathcal{H}) = \mathbb{E}\Big[\big(\mathbb{E}[Y \mid \mathcal{H}] - \mathbb{E}[Y \mid \mathcal{H}_{-a}]\big)^2\Big]\,. \tag{15}$$

Hence, $\Delta_a > 0$ if and only if the candidate source changes the optimal conditional-mean predictor on a set of nonzero probability.

Proof. Let

$$m = \mathbb{E}[Y \mid \mathcal{H}], \qquad m_{-a} = \mathbb{E}[Y \mid \mathcal{H}_{-a}].$$

Under squared loss, the conditional expectation is the optimal predictor. Therefore,

$$\begin{aligned} R(\mathcal{H}_{-a}) &= \mathbb{E}\big[(Y - m + m - m_{-a})^2\big] \\ &= \mathbb{E}[(Y-m)^2] + \mathbb{E}[(m-m_{-a})^2] \\ &\quad + 2\mathbb{E}[(Y-m)(m-m_{-a})]. \end{aligned} \tag{16}$$

Since $m - m_{-a}$ is measurable with respect to $\mathcal{H}$ and $\mathbb{E}[Y - m \mid \mathcal{H}] = 0$, the cross term is zero:

$$\begin{aligned} \mathbb{E}[(Y-m)(m-m_{-a})] &= \mathbb{E}[(m-m_{-a})\mathbb{E}[Y-m \mid \mathcal{H}]] \\ &= 0. \end{aligned} \tag{17}$$

Thus,

$$R(\mathcal{H}_{-a}) = R(\mathcal{H}) + \mathbb{E}[(m-m_{-a})^2],$$

which gives Equation 15. In particular, $\Delta_a = 0$ if and only if $\mathbb{E}[Y \mid \mathcal{H}] = \mathbb{E}[Y \mid \mathcal{H}_{-a}]$ almost surely. □

Subsystem-level predictive support. We first apply Proposition 1 to the latent subsystem dynamics in Equation 6. For subsystem k, define the full lagged latent context

$$\mathcal{H}^z_{k,t} = \left\{\mu^z_{r,t-s} : r = 1, \ldots, K_{\max},\ s = 1, \ldots, p_z\right\}. \tag{18}$$

For a candidate lag-specific source (m, ℓ), let

$$\mathcal{H}^z_{k,t,-(m,\ell)} = \mathcal{H}^z_{k,t} \setminus \{\mu^z_{m,t-\ell}\}. \tag{19}$$

The corresponding population risk gap is

$$\Delta^{z,\ell}_{m\to k} = R(\mathcal{H}^z_{k,t,-(m,\ell)}) - R(\mathcal{H}^z_{k,t}). \tag{20}$$

By Proposition 1,

$$\Delta^{z,\ell}_{m\to k} = \mathbb{E}\Big[\big(\mathbb{E}[\mu^z_{k,t} \mid \mathcal{H}^z_{k,t}] - \mathbb{E}[\mu^z_{k,t} \mid \mathcal{H}^z_{k,t,-(m,\ell)}]\big)^2\Big]. \tag{21}$$

We therefore define the subsystem-level predictive support for target k as

$$\mathcal{S}^z_k = \left\{(m, \ell) : \Delta^{z,\ell}_{m\to k} > 0\right\}. \tag{22}$$

An element $(m, \ell) \in \mathcal{S}^z_k$ means that $\mu^z_{m,t-\ell}$ provides predictive information about $\mu^z_{k,t}$ that cannot be recovered from the remaining latent history. This is the population-level predictive relation that the lag-specific gate $G^{z,\ell}_{mk}$ in Equation 6 is intended to represent.

Variable-level conditional predictive support. The variable graph has a different interpretation because it is learned after accounting for both the target's own history and the shared subsystem context. For target variable i, define its target-excluded subsystem context as

$$c^{\mathrm{sub}}_{i,t} = \sum_k A_{ik} z^{(-i)}_{k,t}, \tag{23}$$

consistent with Equation 7. The full variable-level predictive context is

$$\mathcal{H}^x_{i,t} = \Big(x_{i,t-1:t-p_x},\ c^{\mathrm{sub}}_{i,t},\ \{x_{r,t-s} : r \neq i,\ s = 1, \ldots, p_x\}\Big). \tag{24}$$

For a candidate lag-specific source (j, ℓ), define

$$\mathcal{H}^x_{i,t,-(j,\ell)} = \mathcal{H}^x_{i,t} \setminus \{x_{j,t-\ell}\}. \tag{25}$$

Its population risk gap is

$$\Delta^{x,\ell}_{j\to i} = R(\mathcal{H}^x_{i,t,-(j,\ell)}) - R(\mathcal{H}^x_{i,t}). \tag{26}$$

Again, Proposition 1 gives

$$\Delta^{x,\ell}_{j\to i} = \mathbb{E}\Big[\big(\mathbb{E}[x_{i,t} \mid \mathcal{H}^x_{i,t}] - \mathbb{E}[x_{i,t} \mid \mathcal{H}^x_{i,t,-(j,\ell)}]\big)^2\Big]. \tag{27}$$

The variable-level predictive support is therefore

$$\mathcal{S}^x_i = \left\{(j, \ell) : \Delta^{x,\ell}_{j\to i} > 0\right\}. \tag{28}$$

Importantly, $\Delta^{x,\ell}_{j\to i}$ is conditional on the target's own history, the target-excluded subsystem context, and the histories of the other candidate variables. Thus, an element $(j, \ell) \in \mathcal{S}^x_i$ represents an additional lagged predictive effect of variable j on variable i after shared subsystem variation has already been accounted for. This is the population-level predictive interpretation of the lag-specific cross-variable pathway parameterized by $G^{x,\ell}_{ji}$ in Equation 7. It also distinguishes the role of the two graphs: G^z captures predictive dependence between subsystem states, whereas G^x captures residual local predictive dependence between variables conditional on the shared subsystem context.

Scope of the result. The support sets $\mathcal{S}_k^z$ and $\mathcal{S}_i^x$ are determined by the population distribution once the representation, memberships, and predictive contexts are fixed. In this sense, Equations 22 and 28 define identifiable population predictive supports. The result does not imply that the learned scores $S_{mk}^{z,\ell} = \pi_{mk}^{z,\ell}$ and $S_{ji}^{x,\ell}$ are themselves consistent estimators of these supports under arbitrary model classes, finite samples, or optimization procedures.

A stronger structural or causal interpretation requires additional assumptions. The learned subsystem representation should preserve the causally relevant state up to an admissible, graph-preserving reparameterization. The conditioning context should contain sufficient information for the relation of interest, and the normal mechanism should remain stable over the regime used for learning. In addition, each structural parent must make a nonzero conditional predictive contribution after the other candidate inputs are observed. These assumptions are stronger than those required for the predictive-support result above and are related to the broader identifiability issues studied in causal representation learning (Schölkopf et al., 2021; Yao et al., 2021; Li et al., 2025b).

B Variable Mechanism Parameterization

Each latent edge is modeled by a residual MLP with hidden width $H_z = 32$:

$$\phi_{mk}^{z,\ell}(u) = w_{mk}^{z,\ell} u + \sum_{h=1}^{H_z} v_{mkh}^{z,\ell} \,\mathrm{GELU}(a_{mkh}^{z,\ell} u + c_{mkh}^{z,\ell}). \tag{29}$$

The self and subsystem branches each use two GELU hidden layers of width 32 followed by a scalar output. Each cross-variable edge uses a two-layer residual MLP with the same width. Suppressing edge indices on its parameters,

$$\begin{aligned} \mathbf{h}_1(u) &= \mathrm{GELU}(\mathbf{a}u + \mathbf{c}_1), \qquad \mathbf{h}_2(u) = \mathrm{GELU}(B\mathbf{h}_1(u) + \mathbf{c}_2), \\ \phi_{ji}^{x,\ell}(u) &= wu + \mathbf{v}^\top \mathbf{h}_2(u). \end{aligned} \tag{30}$$

Source and target structural embeddings $\mathbf{u}_j^\ell, \mathbf{v}_i^\ell \in \mathbb{R}^r$ parameterize the lag-specific edge inclusion probability, with target bias b_i^ℓ:

$$\rho_{ji}^{x,\ell} = \sigma\left(\frac{(\mathbf{u}_j^\ell)^\top \mathbf{v}_i^\ell}{\sqrt{r}} + b_i^\ell\right), \qquad j \neq i. \tag{31}$$

During training, $G_{ji}^{x,\ell}$ is a relaxed Bernoulli gate parameterized by $\rho_{ji}^{x,\ell}$; at evaluation, $G_{ji}^{x,\ell} = \rho_{ji}^{x,\ell}$. Together with the soft co-membership C_{ji}, the cross-variable branch is

$$f_i^{\mathrm{cross}} = \sum_{\ell=1}^{p_x} \sum_{j \neq i} C_{ji} G_{ji}^{x,\ell} \phi_{ji}^{x,\ell}(x_{j,t-\ell}). \tag{32}$$

For graph recovery, the path score measures the combined strength of the linear and nonlinear input-to-output paths. It is restricted to hard active-slot graph support and normalized within each target, with numerical floor $\epsilon_q > 0$:

$$\begin{aligned} q_{ji}^\ell &= \mathbb{I}[\widehat{b}_E(j) = \widehat{b}_E(i)]\mathbb{I}[j \neq i] \sqrt{w^2 + \sum_{o,h} (v_o B_{oh} a_h)^2}, \\ S_{ji}^{x,\ell} &= \frac{q_{ji}^\ell}{\max\{\max_{\ell',j'} q_{j'i}^{\ell'}, \epsilon_q\}}. \end{aligned} \tag{33}$$

Here, h and o index the first and second layer hidden units, respectively. The normalization makes the strongest candidate path for each target have score one whenever its unnormalized strength exceeds ϵ_q.

C Complete Training Objective

The full training objective expands the terms summarized in Equation 8. The shared variable-specific decoder is $\mathrm{Dec}_i(\mathbf{z}_t, E) = b_i^d + \sum_k A_{ik}\gamma_k w_{ik}^d z_{k,t}$. Equation 36 defines the three prediction and reconstruction losses. The remaining term Ω contains posterior, hierarchy, graph, and path regularization. With expectations over normal training windows,

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{d}\mathbb{E}\|\mathbf{x}_t - \mathrm{Dec}(\mathbf{z}_t, E)\|_2^2, \tag{34}$$

$$\mathcal{L}_{x} = \frac{1}{d}\mathbb{E}\|\mathbf{x}_t - \widehat{\mathbf{x}}_t\|_2^2, \tag{35}$$

$$\mathcal{L}_{z\to x} = \frac{1}{d}\mathbb{E}\|\mathbf{x}_t - \mathrm{Dec}(\widehat{\mathbf{z}}_t, E)\|_2^2. \tag{36}$$

Each expectation averages over normal windows and the time points available to its loss; the dimensional factors yield elementwise mean squared errors. The full objective is

$$\begin{aligned}\mathcal{L} = {} & \lambda_x\mathcal{L}_x + \lambda_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + \lambda_{z\to x}\mathcal{L}_{z\to x} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}} \\ & + \lambda_{\mathrm{assign}}\mathcal{R}_{\mathrm{assign}} + \lambda_{\mathrm{slot}}\mathcal{R}_{\mathrm{slot}} + \lambda_{\mathrm{ref}}\mathcal{R}_{\mathrm{ref}} + \lambda_{G^z}\mathcal{R}_{G^z} \\ & + \lambda_{G^x}\mathcal{R}_{G^x} + \lambda_{\mathrm{path}}\mathcal{R}_{\mathrm{path}}.\end{aligned} \tag{37}$$

The reported experiments use

$$\begin{gathered}\lambda_x = 1, \quad \lambda_{\mathrm{rec}} = 0.5, \quad \lambda_{z\to x} = 0.5, \quad \lambda_{\mathrm{KL}} = 0.01, \\ \lambda_{\mathrm{assign}} = 0.05, \quad \lambda_{\mathrm{slot}} = 0.2, \quad \lambda_{\mathrm{ref}} = 0.2, \quad \lambda_{G^z} = 0.01, \\ \lambda_{G^x} = 0.001, \quad \lambda_{\mathrm{path}} = 0.005.\end{gathered} \tag{38}$$

The engineering-reference regularizer $\mathcal{R}_{\mathrm{ref}}$ is defined in Equation 10.

The remaining regularizers are summarized below. Let

$$m_k = \frac{1}{d}\sum_i A_{ik}, \qquad \omega_k = \frac{K_{\max} m_k \gamma_k}{\sum_r m_r \gamma_r}, \tag{39}$$

and define $q_{k,t} = \mathcal{N}(\mu_{k,t}^z, v_{k,t}^z)$ and $p_{k,t} = \mathcal{N}(\widehat{z}_{k,t}, \sigma_k^2)$, where σ_k^2 is the learned transition variance of slot k. The posterior, assignment, and slot regularizers are

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{K_{\max}}\mathbb{E}\sum_{k=1}^{K_{\max}} \omega_k \,\mathrm{KL}\left(q_{k,t} \,\|\, p_{k,t}\right), \tag{40}$$

$$\mathcal{R}_{\mathrm{assign}} = -\frac{1}{d}\sum_{i=1}^{d}\sum_{k=1}^{K_{\max}} A_{ik}\log A_{ik}, \tag{41}$$

$$\mathcal{R}_{\mathrm{slot}} = \sum_{k=1}^{K_{\max}} m_k(1-\gamma_k)^2. \tag{42}$$

For the graph regularizers, let $r_i = \sigma(\eta_i)$, $\pi_0^z = 0.25$, and $Z_C = \sum_i \sum_{j\neq i} C_{ji}$. We use

$$\mathcal{R}_{G^z} = \frac{1}{p_z K_{\max}^2}\sum_{\ell,m,k} \mathrm{KL}\left(\mathrm{Bern}(\pi_{mk}^{z,\ell}) \,\|\, \mathrm{Bern}(\pi_0^z)\right), \tag{43}$$

$$\begin{aligned}\mathcal{R}_{G^x} = {} & \frac{1}{Z_C}\sum_{\ell,i}\sum_{j\neq i} C_{ji}\,\mathrm{KL}\left(\mathrm{Bern}(\rho_{ji}^{x,\ell}) \,\|\, \mathrm{Bern}(r_i)\right) \\ & + \frac{0.01}{d}\sum_i \left[-\log r_i - 7\log(1-r_i)\right],\end{aligned} \tag{44}$$

$$\mathcal{R}_{\mathrm{path}} = \frac{1}{Z_C}\sum_{\ell,i}\sum_{j\neq i} C_{ji}\left\|\begin{bmatrix}\mathbf{a}_{ji}^{x,\ell} \\ w_{ji}^{x,\ell}\end{bmatrix}\right\|_2. \tag{45}$$

The graph terms favor sparse edge inclusion, while $\mathcal{R}_{\mathrm{path}}$ promotes sparse predictive paths (Tank et al., 2021; Louizos et al., 2017).

D TRAINING, CALIBRATION, AND NUMERICAL DETAILS

The model uses subsystem lag $p_z = 1$ and rank $r = 8$ for the factorized variable-edge logits. The variable lag is $p_x = 1$. Optimization uses AdamW, gradient clipping, temperature annealing, and phase-wise early stopping. Training follows the same three stages. Stage 1 performs representation warm-up with dense graph supports. Stage 2 learns the latent transition and subsystem graph. Stage 3 enables variable-edge learning and jointly refines the predictive model. At each phase transition, we restore the checkpoint with the lowest normal validation loss from the preceding phase before continuing optimization.

For residual calibration in Section 3.5, the window magnitude is

$$a_{i,n} = \frac{1}{|\mathcal{T}_n|} \sum_{t \in \mathcal{T}_n} |\widetilde{e}_{i,t}|. \tag{46}$$

Let Φ denote the standard normal cumulative distribution function. The robust center and Gaussian-consistent MAD scale are

$$m_i = \mathrm{median}_0(e_{i,t}), \qquad d_i = \frac{\mathrm{median}_0 \, |e_{i,t} - m_i|}{\Phi^{-1}(3/4)}. \tag{47}$$

The factor $[\Phi^{-1}(3/4)]^{-1}$ is the Gaussian consistency correction for the median absolute deviation.

If the robust scale is numerically degenerate, we first fall back to the standard deviation of the normal residuals and then to a unit scale if necessary. We fit a generalized Pareto distribution to window-level magnitudes above the empirical 0.95 quantile when at least 20 exceedances are available and the fit is valid. For the remaining cases, we use the empirical survival function for calibration. Benjamini–Hochberg screening uses an FDR level of 0.01 for TEP and 0.05 for the other datasets. If no subsystem passes the screening step, we select the subsystem with the smallest calibrated p-value as the candidate for Stage 2.

For Equation 5, define the remaining peer mass after excluding variable i as $m_k^{(-i)} = \sum_j A_{jk} - A_{ik}$. When $m_k^{(-i)}$ exceeds a small numerical tolerance, we use Equation 5 directly. If the target is effectively the only member of a slot, we instead use the full pooled aggregate $\mu_{k,t}^z$ from Equation 4 as a numerical fallback for the subsystem-context branch. This fallback includes the target in the pooled value, so strict target exclusion holds only when sufficient peer mass remains.

For graph F1 and HD, we first aggregate lag-specific scores by taking their maximum over lags. We then binarize the continuous scores using the same tail-gap rule for all methods. Candidate quantiles are searched over upper tails containing 2–50% of eligible edges. We calibrate within-subsystem variable graphs separately from subsystem graphs. AUC-PR and AUC-ROC use the unthresholded continuous scores.

E EVALUATION METRICS

E.1 GRAPH RECOVERY

Let $\mathcal{E}_g$ denote the set of off-diagonal candidate edges after collapsing lag-specific scores. With edge precision Prec and recall Rec, the graph-recovery metrics are

$$\mathrm{F1} = \frac{2 \, \mathrm{Prec} \, \mathrm{Rec}}{\mathrm{Prec} + \mathrm{Rec}}, \qquad \mathrm{HD} = \frac{1}{|\mathcal{E}_g|} \sum_{e \in \mathcal{E}_g} \mathbb{I}[\widehat{A}_e \neq A_e]. \tag{48}$$

HD is normalized by the number of evaluated candidate edges. AUC-PR and AUC-ROC are computed from the continuous edge scores. For within-subsystem variable graphs, eligible edges are pooled across subsystems before computing the metrics.

E.2 VARIABLE LOCALIZATION

We adopt the core AC scoring rule of AERCA (Han et al., 2025) under our unified evaluation protocol. Let $\mathcal{E}_r$ denote the set of test events, and let $\mathcal{T}_e^+$ denote the valid evaluation positions with

root-cause labels for event e. For each $t \in \mathcal{T}_e^+$, let $\mathcal{R}_{e,t}$ be the set of labelled root variables and let $\mathrm{Top}_k(e,t)$ denote the top-k variables ranked at that position. We define

$$\mathrm{AC@}k = \frac{1}{|\mathcal{E}_r|} \sum_{e \in \mathcal{E}_r} \frac{1}{|\mathcal{T}_e^+|} \sum_{t \in \mathcal{T}_e^+} \frac{|\mathcal{R}_{e,t} \cap \mathrm{Top}_k(e,t)|}{\min(k, |\mathcal{R}_{e,t}|)}, \tag{49}$$

and

$$\mathrm{Avg@}K = \frac{1}{K} \sum_{k=1}^{K} \mathrm{AC@}k. \tag{50}$$

Thus, AC evaluates variable localization separately at each labelled evaluation position, averages the scores within an event, and then averages across events. Evaluation positions without root-cause labels do not contribute to AC.

E.3 Time-variable localization

We also adopt the core AC* scoring rule of AERCA (Han et al., 2025). Let $\mathcal{T}_e$ denote all valid evaluation positions in the predefined evaluation range of event e. The candidate time-variable set is

$$\mathcal{C}_e^* = \{(t,i) : t \in \mathcal{T}_e,\ i = 1, \ldots, d\}. \tag{51}$$

Let $\mathcal{R}_e^* \subseteq \mathcal{C}_e^*$ denote the labelled root time-variable pairs, and let $\mathrm{Top}_k^*(e)$ denote the top-k pairs ranked within $\mathcal{C}_e^*$ using the scores provided by each method. We define

$$\mathrm{AC^*@}k = \frac{1}{|\mathcal{E}_r|} \sum_{e \in \mathcal{E}_r} \frac{|\mathcal{R}_e^* \cap \mathrm{Top}_k^*(e)|}{\min(k, |\mathcal{R}_e^*|)}, \tag{52}$$

and

$$\mathrm{Avg^*@}K = \frac{1}{K} \sum_{k=1}^{K} \mathrm{AC^*@}k. \tag{53}$$

Unlike AC, AC* ranks time-variable candidates jointly across the event range. Therefore, high scores at non-root positions can displace true root pairs from the top-k. This metric evaluates joint time-variable localization within a predefined event range rather than fault detection over an unconstrained test sequence.

F Real-data Analysis of the Two-Stage Inference Procedure

We further compare two-stage inference with direct ranking on the four real datasets under the same evaluation protocol. Both settings use the same trained hierarchical predictor and calibrated innovation scores, and differ only in whether candidate-subsystem screening is used before ranking. Table 6 reports variable-ranking results, while Table 7 reports joint time-variable-ranking results.

The effect of two-stage inference varies across datasets and metrics. For variable ranking, two-stage inference improves Avg@10 from 0.870 to 0.877 on Probe and from 0.435 to 0.472 on TEP, matches direct ranking at 0.839 on Coolant, and is lower on Hydraulics (0.680 versus 0.759). For time-variable ranking, the two approaches are identical on Hydraulics and TEP and differ only slightly on Coolant and Probe.

G Limitations

A main limitation of HiRCA is its reliance on an engineering subsystem prior. Although the membership structure is optimized through soft assignments, the learned hierarchy is still initialized and regularized by the provided engineering partition. When this prior is accurate, as in the synthetic experiments, the hierarchical decomposition provides a reliable basis for subsystem screening and coarse-to-fine localization. In real systems, however, engineering partitions may not exactly match the dependency structure relevant to fault propagation and root cause localization. Such mismatch can reduce the effectiveness of the two-stage inference procedure.

Table 6: Real-data comparison of two-stage inference and direct ranking: variable ranking.

Dataset	Model	AC@1↑	AC@3↑	AC@5↑	AC@10↑	Avg@10↑
Coolant	Direct ranking	$0.722_{\pm 0.011}$	$0.651_{\pm 0.005}$	$0.785_{\pm 0.005}$	$1.000_{\pm 0.000}$	$0.839_{\pm 0.002}$
	HiRCA	$0.721_{\pm 0.011}$	$0.650_{\pm 0.006}$	$0.785_{\pm 0.005}$	$1.000_{\pm 0.000}$	$0.839_{\pm 0.002}$
Hydraulics	Direct ranking	$0.268_{\pm 0.000}$	$0.580_{\pm 0.000}$	$0.838_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.759_{\pm 0.000}$
	HiRCA	$0.172_{\pm 0.000}$	$0.339_{\pm 0.000}$	$0.838_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.680_{\pm 0.000}$
Probe	Direct ranking	$0.682_{\pm 0.023}$	$0.749_{\pm 0.002}$	$0.816_{\pm 0.019}$	$1.000_{\pm 0.000}$	$0.870_{\pm 0.006}$
	HiRCA	$0.683_{\pm 0.023}$	$0.751_{\pm 0.007}$	$0.876_{\pm 0.061}$	$1.000_{\pm 0.000}$	$0.877_{\pm 0.023}$
TEP	Direct ranking	$0.225_{\pm 0.018}$	$0.388_{\pm 0.017}$	$0.466_{\pm 0.019}$	$0.524_{\pm 0.016}$	$0.435_{\pm 0.013}$
	HiRCA	$0.231_{\pm 0.018}$	$0.423_{\pm 0.017}$	$0.509_{\pm 0.015}$	$0.564_{\pm 0.011}$	$0.472_{\pm 0.010}$

Table 7: Real-data comparison of two-stage inference and direct ranking: time-variable ranking.

Dataset	Model	AC*@1↑	AC*@10↑	AC*@100↑	AC*@500↑	Avg*@500↑
Coolant	Direct ranking	$0.816_{\pm 0.134}$	$0.932_{\pm 0.017}$	$0.696_{\pm 0.025}$	$0.500_{\pm 0.005}$	$0.590_{\pm 0.004}$
	HiRCA	$0.792_{\pm 0.140}$	$0.871_{\pm 0.089}$	$0.695_{\pm 0.026}$	$0.536_{\pm 0.005}$	$0.577_{\pm 0.015}$
Hydraulics	Direct ranking	$0.182_{\pm 0.000}$	$0.200_{\pm 0.000}$	$0.330_{\pm 0.000}$	$0.291_{\pm 0.000}$	$0.303_{\pm 0.000}$
	HiRCA	$0.182_{\pm 0.000}$	$0.200_{\pm 0.000}$	$0.330_{\pm 0.000}$	$0.291_{\pm 0.000}$	$0.303_{\pm 0.000}$
Probe	Direct ranking	$0.353_{\pm 0.487}$	$0.792_{\pm 0.101}$	$0.589_{\pm 0.003}$	$0.686_{\pm 0.001}$	$0.641_{\pm 0.010}$
	HiRCA	$0.353_{\pm 0.487}$	$0.792_{\pm 0.101}$	$0.589_{\pm 0.003}$	$0.686_{\pm 0.025}$	$0.640_{\pm 0.014}$
TEP	Direct ranking	$0.000_{\pm 0.000}$	$0.007_{\pm 0.009}$	$0.132_{\pm 0.006}$	$0.141_{\pm 0.004}$	$0.127_{\pm 0.005}$
	HiRCA	$0.000_{\pm 0.000}$	$0.007_{\pm 0.009}$	$0.132_{\pm 0.006}$	$0.141_{\pm 0.004}$	$0.127_{\pm 0.005}$

An important direction for future work is therefore to reduce the dependence on predefined engineering partitions. One possibility is to learn subsystem structure directly from multivariate time-series data, while using engineering knowledge only as an optional weak constraint when available. Another direction is to jointly infer subsystem assignments and causal dependencies in a fully data-driven manner. These extensions could improve the robustness of hierarchical RCA when reliable engineering priors are unavailable or only partially correct.