Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

EvoCausalTwin: Drift-Aware Causal Structure Learning, Evidence-Calibrated Verification, and Budget-Constrained Intervention for Evolving Systems

Phat Hai Tien Nguyen

phathanos0907@gmail.com

Ton Duc Thang University

Research Article

Keywords:

Posted Date: November 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11668674/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

EvoCausalTwin: Drift-Aware Causal Structure Learning, Evidence-Calibrated Verification, and Budget-Constrained Intervention for Evolving Systems

Phat Hai Tien Nguyen

Department of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam

Email: phathanos0907@gmail.com ORCID: 0009-0006-9059-9345

Corresponding author

Abstract

Evolving systems undergo distributional, mechanistic, and structural changes that can invalidate static causal models and lead to poorly justified interventions. This paper presents EvoCausalTwin, an evidence-gated artificial intelligence framework linking dynamic causal structure learning, causal-drift decomposition, evidence-calibrated verification, budget-aware intervention, causal-effect decision analysis, and constrained workflow auditing. Its three language-model auditor roles share one quantized Llama-3.1-8B-Instruct model and cannot orient graph edges or bypass statistical verification. We evaluate the framework on EvoCausalBench, comprising 2,000 source structural causal models and 50,000 episodes across four graph families, five mechanism families, five drift regimes, and 10–100 variables. Splits are assigned at the source-SCM level before episode generation. On the principal test set, EvoCausalTwin obtains SHD 4.8, directed F1 0.79, ATE RMSE 0.22, PEHE 0.32, five-class drift macro-F1 0.77, decision regret 0.24, and drift-class ECE 0.061, compared with 6.1, 0.74, 0.27, 0.39, 0.66, 0.36, and 0.142 for CD-NOD. Paired analysis uses 300 held-out SCM-level vectors. A separate 50,000-observation reliability audit yields decision-confidence weighted ECE 0.01845, with subgroup calibration reported across all five drift regimes. A full 2^4 factorial ablation distinguishes the descriptive contributions of modules M2–M5. Twelve-month operational evidence is retrospective and does not establish prospective causal treatment effectiveness. The findings support a framework-level contribution under the evaluated protocols, not a new standalone causal-discovery estimator.

Keywords: causal discovery; non-stationarity; mechanism drift; active intervention; uncertainty calibration; causal benchmarking

1 Introduction

Causal discovery seeks a graph or equivalence class that represents the data-generating relations among variables. In many operational systems, however, the data-generating process is not stationary over the full period of interest. A process may move between operating modes, experience sensor recalibration, undergo changes in control policy, or acquire a genuinely new causal dependency. The practical challenge is not merely to detect that the distribution

has changed. It is to determine what changed, assess whether the currently inferred graph remains defensible, and decide whether a costly intervention or model update is warranted.

Three change types are particularly important. Covariate shift changes the distribution of observed variables while leaving the structural assignments invariant. Mechanism drift changes one or more structural assignments while retaining the graph support. Structural drift changes the edge set itself. Compound drift combines two or more of these effects. These categories are conceptually distinct and motivate separate detection and verification pathways. The distinction is consistent with work on heterogeneous and non-stationary causal discovery, including CD-NOD, which uses distributional heterogeneity as information for causal structure estimation [10, 21].

A static causal learner can be reliable within a stable regime and still become misleading after a regime transition. Conversely, rebuilding a graph whenever a monitoring statistic changes can waste computation and intervention budget. A robust operational system therefore needs a sequence of linked decisions: monitor the stream, decompose the evidence for drift, update the candidate causal structure, verify assumptions and uncertainty, choose interventions under a budget, and evaluate the expected causal and operational consequences. The design also needs an explicit abstention option when evidence is insufficient.

This paper introduces EvoCausalTwin, a framework organized around six modules: (M1) dynamic causal structure learning, (M2) causal drift decomposition, (M3) evidence-calibrated causal verification, (M4) budget-aware active intervention, (M5) causal effect and decision engine, and (M6) a three-role workflow auditor. The modules are not interchangeable. M1 estimates graph structure using statistical causal discovery. M2 classifies the type of change. M3 assesses whether graph and effect claims are supported. M4 selects information-gathering or operational interventions under a finite budget. M5 translates verified causal estimates into decision scores. M6 processes metadata and assumption/context evidence, but all assertions that could affect the graph must pass the statistical evidence gate.

1.1 Problem statement

Let $X_t = (X_{1t}, \ldots, X_{pt})$ denote the observed variables at time t, and let $G_t = (V, E_t)$ denote the causal graph associated with the current regime. A structural causal model may be written as

$$X_{it} = f_{i,t}(X_{\mathrm{Pa}_{G_t}(i),t}, U_{it}), \qquad i = 1, \ldots, p, \tag{1}$$

where $f_{i,t}$ is the structural assignment, $\mathrm{Pa}_{G_t}(i)$ is the parent set, and U_{it} collects exogenous influences. The operational task is to infer whether a change between windows W_{t-1} and W_t is primarily distributional, mechanistic, structural, or compound, and then select a safe update or intervention.

We distinguish a distributional change in $P_t(X)$ from a change in $f_{i,t}$ and from a change in E_t. This distinction matters because a shift in $P(X)$ can occur with invariant structural assignments, and a changed mechanism can occur without changing the graph skeleton. The system must also account for uncertainty in edge existence, orientation, effect estimates, and the action utility. We therefore formulate the output as a tuple comprising a candidate graph, drift class, evidence record, calibrated confidence, causal effect estimates, selected action or abstention, and an audit trail.

1.2 Contributions and novelty

The contribution is a new evidence-constrained framework and evaluation protocol, not a claim that the individual estimators, drift statistics, calibration procedures, or LLM backbone are

novel algorithms. The methodological novelty lies in the specified interaction contract: drift decomposition determines which update hypotheses are considered; M3 gates graph/effect assertions; M4 selects only actions that satisfy budget and risk constraints; M5 translates verified effects into utility; and M6 can contribute context only through M3. This is a stronger and more falsifiable claim than describing the system as a loose aggregation of modules.

1. **Drift-decomposed causal learning.** We integrate regime monitoring with separate covariate, mechanism, structural, and compound drift hypotheses. The empirical contribution is a unified evaluation across five drift classes and four graph sizes, with graph, effect, and drift metrics reported separately.
2. **Decision-aware intervention under explicit budget.** We connect causal uncertainty and estimated utility to an active intervention policy. The contribution is assessed through cost per improvement, regret, negative-utility frequency, and an oracle decomposition rather than through graph accuracy alone.
3. **Evidence-calibrated verification and abstention.** We evaluate reliability using ECE, Brier score, reliability slope/intercept, coverage-risk curves, and unresolved-edge coverage. The intended novelty is the operational coupling of calibration and abstention to causal decision-making, not a new general-purpose calibration theorem.
4. **EvoCausalBench with hierarchical ground truth.** We provide a benchmark protocol spanning 2,000 SCMs, four graph families, five mechanism families, five drift regimes, and 10/20/50/100-variable settings. A three-level evidence policy prevents full-graph SHD claims when only expert-reference or event-level labels are available.
5. **Constrained three-role workflow auditing.** We evaluate metadata interpretation, assumption auditing, and context-anomaly verification as separate roles implemented with one shared Llama-3.1-8B-Instruct Q4_K_M model. The ablation isolates role separation from model diversity and confirms that the roles do not improve SHD in the reported experiment.

The main novelty claim is therefore that EvoCausalTwin combines these functions into an auditable drift-to-decision pipeline and evaluates their joint operational behavior. We do not claim that language models discover causal edges, that the system solves causal identifiability in arbitrary settings, or that the reported synthetic benchmark results guarantee performance in every deployment domain.

1.3 Research questions and hypotheses

Five hypotheses structure the evaluation. H1 asks whether drift decomposition distinguishes five drift regimes better than the strongest reported baseline. H2 asks whether adaptive learning improves structural and effect estimation after regime changes. H3 asks whether decision-aware intervention reduces regret and resource cost. H4 asks whether calibration and abstention reduce high-confidence decision risk. H5 asks whether the three-role auditor improves assumption and context assessment without replacing statistical causal discovery. The hypotheses are evaluated using the reported test metrics and ablations; the LLM hypothesis is treated as conditional because the graph SHD is unchanged across LLM configurations.

1.4 Paper organization

Section 2 reviews the methodological background. Section 3 presents the architecture and its modules. Section 4 formalizes the learning, drift, verification, intervention, and decision components. Section 5 describes the benchmark and experimental protocol. Sections 6–10 report the main, graph-type, drift, intervention, calibration, robustness, generalization, and

real-data results. Section 11 analyzes the workflow auditor and longitudinal evidence. Section 12 discusses limitations and validity boundaries. Section 13 concludes. Appendices provide detailed benchmark, implementation, and reproducibility information.

2 Background and related work

2.1 Causal structure learning

Constraint-based methods infer graph structure from conditional independence relations. The PC algorithm is a canonical example, while FCI extends the search to settings with possible latent confounding and returns a partial ancestral graph rather than a fully oriented DAG [19]. Score-based methods such as GES search over equivalence classes using a score criterion [3]. Differentiable methods such as NOTEARS formulate DAG learning as a continuous optimization problem with an acyclicity constraint [22]. Nonlinear methods, including CAM and LiNGAM-family approaches, make different assumptions about functional form and noise distributions. Temporal methods such as PCMCI+ explicitly account for lagged dependencies and autocorrelation [16, 17].

No single family dominates across all data-generating mechanisms. Conditional-independence tests can become unreliable in high-dimensional settings or under assumption violations. Score-based and differentiable approaches depend on their scoring or optimization assumptions. Nonlinear methods can benefit from functional asymmetries but remain sensitive to misspecification. Temporal methods address lag structure but should not be compared to static DAG methods as though the output spaces were identical. The evaluation in this paper therefore keeps CPDAG, DAG, temporal, and PAG panels separate.

2.2 Non-stationary and heterogeneous causal discovery

Non-stationarity is not simply a nuisance. Changes across environments can help identify causal directions when mechanisms remain invariant in the relevant sense. CD-NOD explicitly addresses causal discovery from heterogeneous and non-stationary data, including mechanism-change detection and orientation [10, 21]. More recent work continues to study causal discovery under non-stationary time series and varying regimes [6, 18]. The EvoCausalTwin design follows this line of research but emphasizes an operational decomposition of the change type and its downstream implications for intervention and decision-making.

2.3 Intervention and causal effect estimation

Causal discovery and causal effect estimation are related but distinct. A graph may suggest adjustment sets or candidate intervention targets, but effect estimation still depends on identification assumptions and estimator quality. Structural causal models provide a formal language for interventions and counterfactual reasoning [12, 14]. In active learning, an intervention can be chosen for information gain, expected utility, or a combination of both. The policy in this paper uses an explicit resource budget and compares random, graph-uncertainty-only, CD-NOD, and EvoCausalTwin policies. Because observational and intervention data can have different identification properties, the paper reports intervention-policy outcomes separately from graph recovery metrics.

2.4 Calibration and selective prediction

Predictive confidence is not equivalent to reliability. Expected calibration error and the Brier score summarize different aspects of probabilistic quality, while reliability slope and intercept expose systematic over- or under-confidence. Modern neural predictors can be miscalibrated even when discrimination is strong [8]. Selective prediction adds an abstention option: a system can decline to make a decision when its estimated risk is high, trading coverage for lower selective risk [7]. EvoCausalTwin applies these principles to drift and causal decision outputs, reporting calibration by task and a coverage-risk curve.

2.5 Language models as workflow auditors

Language models can summarize unstructured context and help surface candidate assumption violations, but text generation alone does not establish a causal relation. The design therefore constrains the LLM roles to metadata interpretation, assumption auditing, and contextual anomaly verification. No role is permitted to create or reverse graph edges. The resulting claims must be checked by M3 using statistical evidence. This is a safety and separation-of-concerns design choice, not a claim that LLMs improve causal discovery accuracy in general.

Table 1 compares the framework with representative methodological families. The comparison is deliberately functional: it identifies the principal objective and the interface that EvoCausalTwin adds, without claiming that prior methods lack every form of monitoring or uncertainty estimation. CD-NOD is the closest non-stationary discovery reference; active discovery methods such as Track-and-Stop optimize intervention allocation for graph identification; and LLM-based systems such as ICDA use language-model uncertainty and intervention feedback to update candidate edges. EvoCausalTwin instead restricts the LLM to non-decisional audit roles and places action utility, evidence calibration, and operational risk in a shared constrained workflow [4, 5, 9, 10, 21].

Table 1 Functional comparison with related causal discovery and decision frameworks

Framework family	Representative methods	Primary objective / output	EvoCausalTwin distinction
Non-stationary causal discovery	CD-NOD	Causal discovery from heterogeneous or non-stationary observations; graph structure under changing distributions	Explicitly decomposes operational drift into covariate, mechanism, structural, and compound labels, then links the label to verification and update control
Active causal discovery	IGSP-style intervention search; Track-and-Stop	Select interventions to identify graph structure with fewer experiments or samples	Scores actions using expected operational utility, intervention cost, and risk constraints; graph identification is not the sole objective
Calibration and selective prediction	ECE/Brier calibration; selective classification	Quantify confidence and abstain when prediction risk is high	Applies the evidence gate and abstention to graph, effect, and action decisions in one workflow
LLM-assisted causal discovery	ICDA and data/LLM hybrid discovery	Use semantic metadata, LLM uncertainty, or intervention feedback to propose/update causal hypotheses	Uses one LLM instance only for metadata, assumption, and context auditing; it cannot add or orient graph edges
EvoCausalTwin	M1–M6	Drift-to-decision framework with benchmark and hierarchical evidence levels	Couples drift decomposition, evidence verification, budgeted action, causal utility, and a constrained workflow auditor; evaluated as a framework rather than a new standalone estimator

2.6 What is and is not claimed as novelty

The comparison in Table 1 motivates three bounded novelty claims. First, M2 does not claim that distributional heterogeneity or non-stationary causal discovery was unknown before CD-NOD. Its contribution is an explicit operational decomposition rule that uses residual-distribution evidence and graph-overlap evidence to distinguish covariate, mechanism, and structural changes, with a compound label when multiple tests trigger. Second, M4 does not claim that active intervention selection is new. Track-and-Stop and related methods optimize intervention efficiency for graph identification; EvoCausalTwin evaluates a different objective in which expected utility is penalized by cost and risk under a finite operational budget. Third, M6 does not claim LLM-based causal discovery. In contrast to methods that allow an LLM to propose or update edges, the auditor is non-decisional: it produces structured context/assumption flags, and M3 alone determines whether any assertion is admissible. These claims are framework-level contributions whose value depends on the measured interaction and ablation results, not on a new theorem for causal identification.

The closest baseline, CD-NOD, already addresses causal discovery under heterogeneous or non-stationary data. Accordingly, EvoCausalTwin is not presented as a replacement estimator that universally dominates CD-NOD under all assumptions. The intended advance is to

connect non-stationary discovery to evidence-calibrated verification, risk-limited interventions, utility-based decisions, and a benchmark protocol that distinguishes synthetic truth from expert and operational references. The main comparison therefore supports the integrated framework only within the tested graph families, mechanisms, drift regimes, and evaluation protocols.

3 EvoCausalTwin architecture

3.1 System overview and information flow

EvoCausalTwin is a sequential, evidence-gated architecture. Streaming observations and operating metadata enter M1, which maintains a candidate causal representation over the current analysis window. M2 compares the current window with the historical reference and separates evidence consistent with covariate shift, mechanism drift, structural drift, or compound drift. The decomposition is not itself an edge-orientation operation; it supplies a regime hypothesis and evidence summary to the subsequent verification stage.

M3 checks the candidate graph and derived claims against statistical evidence. Its responsibilities include residual diagnostics, assumption checks, confidence calibration, and unresolved-edge handling. M4 selects interventions or information-gathering actions subject to a finite budget. M5 estimates causal effects and translates them into decision scores, including expected utility and regret. M6 processes contextual information in three distinct roles and submits any assertion to the same evidence gate. Observed outcomes are fed into subsequent windows, while held-out labels remain isolated for evaluation.

The architecture is deliberately modular. A module may produce a candidate or diagnostic, but no module can silently bypass the verification and safety checks. In particular, M6 cannot orient edges; M4 cannot execute a recommendation solely because an LLM proposes it; and M5 cannot convert an unverified association into a causal effect claim. This separation makes it possible to attribute improvements and failures to the appropriate stage.

Fig. 1 presents the complete system in a staged layout. The upper row follows the seven-step operational path from source SCM observations through dynamic structure learning, drift decomposition, verification, budget-aware intervention, causal decision state, and held-out evaluation. The middle branch expands the three-role workflow auditor and the monitor/controller functions; the lower selection group separates risk screening, budget/utility ranking, and protected abstention. The bottom reference band distinguishes Level A synthetic SCM truth, Level B expert/task references, and Level C operational labels. The held-out-label note makes the evaluation boundary explicit: test labels are reserved for final assessment and do not feed drift learning or intervention selection.

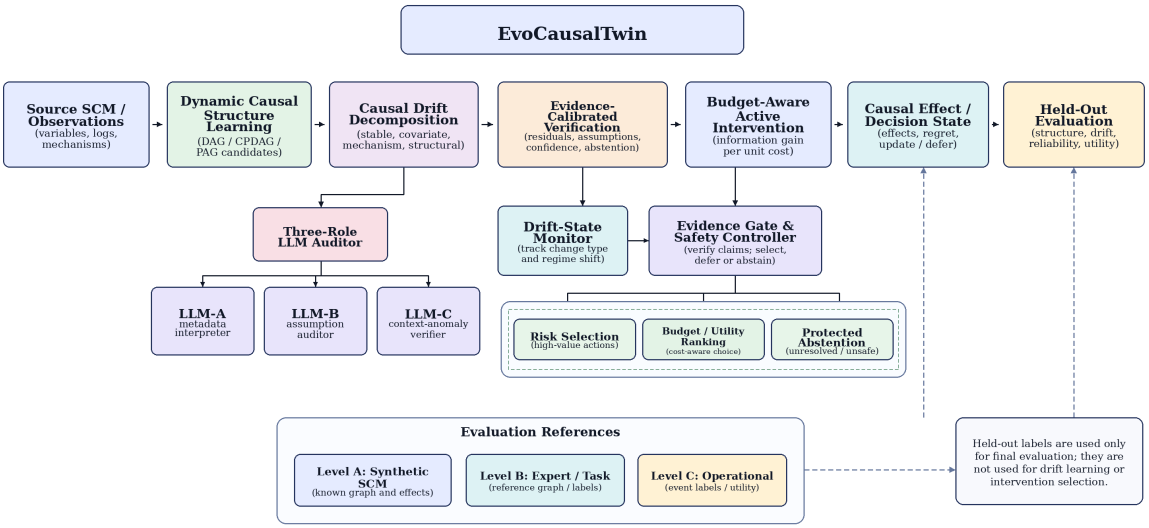


Fig. 1 System architecture of EvoCausalTwin. The upper row defines the seven-stage operational path; lower branches isolate the three-role auditor and the monitor, evidence, budget, and safety controls. The bottom band separates Level A–C evaluation references, while held-out labels remain reserved for final assessment.

3.2 Dynamic causal structure learning (M1)

M1 estimates a causal representation for the current operating window and retains the preceding model as a comparison reference. The implementation is evaluated against multiple causal discovery families rather than a single narrow baseline. The principal synthetic benchmark uses a common preprocessing and tuning protocol, and the method selection is locked using validation data before test evaluation. For static DAG panels, the output is compared against the known synthetic DAG. For CPDAG and PAG panels, the evaluation respects the equivalence-class or endpoint-mark representation. For temporal data, lagged and contemporaneous edges are scored separately.

A useful abstraction is to let $\widehat{G}_t$ denote the current candidate and $\widehat{G}_t - 1$ the previous candidate. The update decision is not triggered by a marginal distribution shift alone. Instead, M1 receives the drift evidence from M2 and the verification status from M3. If the evidence indicates only covariate shift and the causal mechanisms remain compatible with the previous model, the system can retain the graph while recalibrating downstream quantities. If mechanism or structural evidence is stronger, a targeted re-estimation is considered. The implementation is therefore described at the module level; no particular optimizer implementation is assumed beyond the evaluated decision protocol.

3.3 Causal drift decomposition (M2)

M2 classifies five regimes: stable, covariate, mechanism, structural, and compound. Stable indicates no detected regime change. Covariate shift indicates a distributional change with no corresponding evidence that the structural assignments have changed. Mechanism drift indicates a change in one or more assignments while graph support remains stable. Structural drift indicates changed causal adjacency. Compound drift indicates co-occurring change types. The five-class detector is evaluated with macro-F1 and class ECE; binary drift detection is separately evaluated with F1, change-point F1, delay, and false alarms.

Fig. 2 explains the conceptual distinction among these regimes. The figure supports

the interpretation of the five-class results and prevents the common mistake of treating all non-stationarity as a structural graph change.

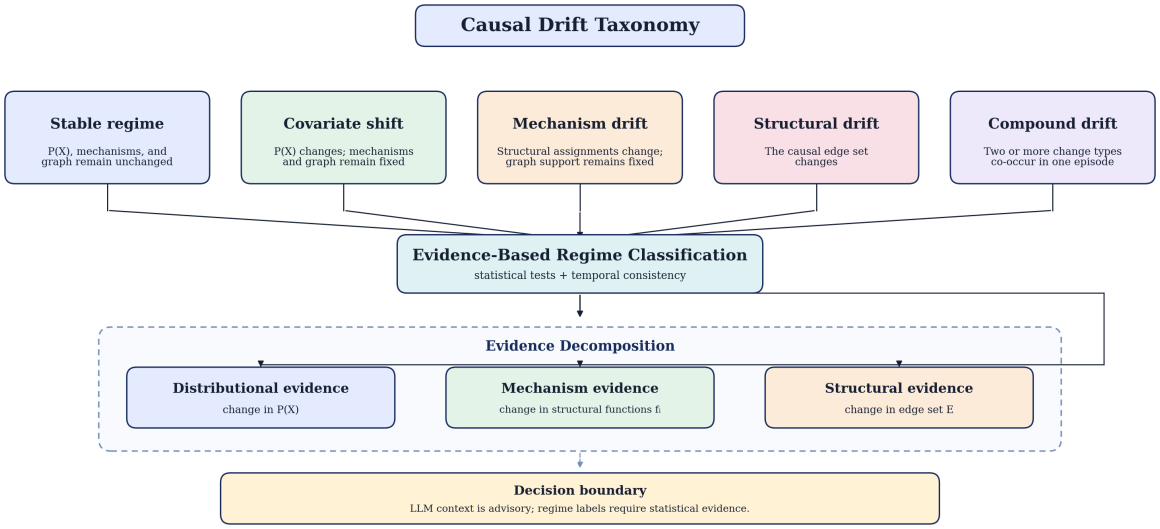


Fig. 2 Operational taxonomy of stable, covariate, mechanism, structural, and compound regimes. The classification describes the type of change, not an automatic graph-edit instruction

The evaluated decomposition rule combines three signals. Let J_t be the Jaccard overlap between the edge sets in consecutive candidate graphs, let D_t be the maximum-mean-discrepancy statistic between current and reference residual distributions, and let p_t^{KS} be the Kolmogorov–Smirnov test p-value for the observed-variable distributions. The recorded thresholds are $J_t < 0.70$, $D_t > 0.20$, and $p_t^{KS} < 0.05$. A structural flag is raised when graph overlap falls below its threshold; a mechanism flag is raised when residual distributions differ; and a covariate flag is raised when the observed-variable distribution changes without a corresponding mechanism or structural flag. If two or more flags are simultaneously active, the regime is recorded as compound rather than forcing a single-cause label. Table 2 lists the three statistics, fixed thresholds, and operational interpretation; its purpose is to make the mapping from observed evidence to a drift label auditable rather than leave the classification as an informal narrative.

Table 2 M2 drift-decomposition decision rule and thresholds

Signal	Operational statistic	Trigger	Interpretation
Structural	Jaccard overlap $J_t = \|E_t - 1 \cap E_t\| / \|E_t - 1 \cup E_t\|$	$J_t < 0.70$	Candidate edge-set change; targeted graph re-estimation is considered
Mechanism	MMD between residual distributions	$D_t > 0.20$	Structural assignments may have changed while graph support is retained
Covariate	KS test on observed-variable windows	$p_t^{KS} < 0.05$	Observed distribution changed; by itself this is not evidence of a graph edit
Compound	Two or more flags active	At least two triggers	Preserve multiple change hypotheses and use conservative verification
Stable	No flag active	No trigger	Retain current graph subject to routine monitoring

M2 pseudocode. The following is the implemented decision order expressed with explicit multi-flag handling; the thresholds are fixed in the reported configuration.

```
Input:  current/reference windows, current/previous graphs, fitted assignments
1.  r_t <- X_t - f-hat(X_t, G_t)
2.  structural <- [Jaccard(E_t-1, E_t) < 0.70]
3.  mechanism <- [MMD(current residuals, reference residuals) > 0.20]
4.  covariate <- [KS-test(X_t, X_t-1) p-value < 0.05]
5.  if count(structural, mechanism, covariate) >= 2:  label COMPOUND
6.  else if structural:  label STRUCTURAL
7.  else if mechanism:  label MECHANISM
8.  else if covariate:  label COVARIATE
9.  else:  label STABLE
Output:  regime label, three flags, and supporting statistics
```

3.4 Evidence-calibrated causal verification (M3)

M3 is the central evidence gate. It receives candidate graph changes, residual evidence, assumption diagnostics, and contextual signals. The gate distinguishes evidence supporting edge existence from evidence supporting orientation, and it tracks unresolved edges rather than forcing a fully directed graph when the available evidence does not justify one. The CPDAG evaluation records micro/macro skeleton and orientation errors, unresolved-edge rate, orientation coverage, and orientation accuracy. The PAG evaluation records endpoint-mark errors and unresolved-edge rates.

Calibration is evaluated separately for drift classes, edge existence, edge orientation, and decision outputs. The reported main benchmark uses validation-defined bins and applies the calibration evaluation to test data. Fifteen equal-frequency bins are used for the Brier/reliability panel. The selective prediction mechanism allows the system to abstain when confidence is insufficient; the coverage-risk curve quantifies the risk reduction across coverage levels. The implementation uses 1,000 bootstrap resamples, significance levels of 0.05/0.10 for the recorded tests, and a conformal level of 0.10 selected from the validation candidates

0.05, 0.10, and 0.15. In the PAG failure mitigation, the workflow abstains when unresolved endpoints exceed 40%.

M3 pseudocode. M3 is an admissibility gate, not a graph generator. It evaluates evidence supplied by M1, M2, and M6 and returns either a verified claim set or an abstention status.

```
Input:  candidate graph, residuals, drift flags, M6 audit assertions
1.  Check graph-representation assumptions (DAG, CPDAG, or PAG)
2.  Run residual and assumption diagnostics on the current window
3.  Bootstrap supported edge/effect summaries (B = 1,000)
4.  Calibrate probabilities using validation-defined bins (15 bins)
5.  Apply conformal acceptance level alpha = 0.10
6.  Reject any M6 assertion that lacks a matching statistical evidence record
7.  If unresolved PAG endpoints exceed 0.40:  return ABSTAIN
8.  Else:  return verified claims with provenance
Output:  verified claims, unresolved set, calibration record, or ABSTAIN
```

3.5 Budget-aware active intervention (M4)

M4 chooses an intervention or information-gathering action under a finite resource budget. The evaluated policy comparison includes random selection, graph-uncertainty-only selection, CD-NOD, EvoCausalTwin, and an oracle reference. The reported columns include total cost, total utility change, cost per improvement, episode failure, negative-utility rate, and unsafe rate. The oracle decomposition crosses graph quality with policy quality to separate gains attributable to a better graph from gains attributable to a better decision policy.

The policy is not assumed to maximize utility without risk. A candidate action must pass the safety and verification requirements, and negative utility is monitored as an explicit failure category. Five of 61 executed interventions have negative utility (8.20%). This is an observed failure rate in the supplied experiment, not a guarantee that future deployments will have the same rate. The decision score is constrained by a finite budget B and a risk ceiling $R_{\max}$:

$$\begin{aligned} \max_{z_a \in \{0,1\}} \quad & \sum_{a \in \mathcal{A}} z_a \, \widehat{\mathbb{E}}[U(a) \mid \mathcal{E}_t] - \lambda_c \sum_a z_a C(a) - \lambda_r \sum_a z_a R(a), \\ \text{subject to} \quad & \sum_a z_a C(a) \le B, \qquad R(a) \le R_{\max} \text{ for each selected } a. \end{aligned} \tag{2}$$

Here z_a indicates whether action a is selected. $R_{\max}$ is an application-level safety constraint, not a universal numerical threshold; the reported experiments evaluate the policy under the recorded safety interlocks and intervention budget.

M4 pseudocode.

```
Input:  verified evidence E_t, candidate actions A, remaining budget B
1.  Remove actions that fail M3 verification or safety interlocks
2.  For each action:  estimate U-hat, cost C, and risk R
3.  Score each action:  U-hat - lambda_c C - lambda_r R
4.  Rank feasible actions by score subject to total cost <= B
5.  Select the highest-ranked feasible action(s); do not exceed B
6.  If no action is admissible or evidence is insufficient:  ABSTAIN
7.  Record predicted utility, realized utility, cost, risk, and reason for
rejection
Output:  selected action or abstention plus an auditable decision record
```

3.6 Causal effect and decision engine (M5)

M5 estimates effects and decision consequences from the verified causal representation. The principal synthetic benchmark reports ATE RMSE and PEHE, while real datasets with non-counterfactual evidence use appropriately named quantities such as PEPE, effect RMSE, Wasserstein distance, and observed utility gain. These metrics remain distinct throughout the evaluation; Level C observational deployment outcomes are not interpreted as actual counterfactual regret.

For a candidate action a, the decision engine conceptually evaluates expected utility under the current evidence and resource constraints. A generic decision score can be written as

$$S(a \mid \mathcal{E}_t) = \widehat{\mathbb{E}}[U(a) \mid \mathcal{E}_t] - \lambda_c C(a) - \lambda_r R(a), \tag{3}$$

where $\mathcal{E}_t$ is the verified evidence, $C(a)$ is resource cost, $R(a)$ is a risk penalty, and the weights encode the operating policy. This equation is a high-level representation of the decision objective, not a claim that the source supplies a unique closed-form optimizer for every application.

3.7 Three-role workflow auditor (M6)

M6 uses one Llama-3.1-8B-Instruct model with Q4_K_M quantization and three role-specific prompts, input schemas, and output schemas. LLM-A interprets sensor metadata, operating mode, and logs; LLM-B audits assumptions such as linearity, stationarity, ignorability, and positivity; LLM-C verifies contextual anomalies and severity. The three roles run concurrently using the same model weights. The reported peak memory is 8.6 GB, which is consistent with one quantized model instance rather than three independent 8B models. Aggregate role compute is reported as 29.7 seconds per episode, while end-to-end wall-clock is 26.4 seconds per episode; the reported token volume is approximately 4,200 tokens per episode.

The role separation is a controlled design choice. It isolates the effect of task decomposition from the effect of model diversity. The three roles are not three distinct LLMs. All role outputs remain advisory and must pass M3. The ablation results are reported in Section 10.

3.8 Data and control boundaries

The system uses historical data for normalization and comparison, and it avoids using future data to compute scaling statistics. The benchmark split is performed at the source-SCM level before episode generation. Drift labels are used for the shared supervised drift detector only; they are not passed to the causal discovery backbone. The held-out test labels are reserved for final evaluation. These boundaries are central to the validity of the reported results and are treated as part of the method rather than as incidental implementation details.

4 Methodology and evaluation metrics

4.1 Structural recovery

For synthetic DAGs with known ground truth, structural error is measured using structural Hamming distance. In this study SHD counts insertions, deletions, and reversals. Directed precision, recall, and F1 provide a complementary view because two methods can have similar SHD but different precision-recall trade-offs. For CPDAGs, skeleton and orientation errors are reported separately at both micro and macro levels. For PAGs, endpoint marks (tail,

arrowhead, circle) are evaluated, and unresolved-edge rate is reported alongside orientation coverage and accuracy. This avoids treating every unresolved edge as an error when the equivalence class does not identify a unique orientation.

4.2 Effect estimation

The main synthetic experiment reports ATE RMSE and PEHE. Outcome z-scores are computed within each SCM using training information only. Real datasets without individual counterfactual ground truth do not use PEHE or ATE RMSE labels. Instead, the experiments report PEPE for held-out perturbation targets, effect RMSE and Pearson correlation for scPerturb, and Wasserstein distances where distributional effect comparisons are appropriate. These metric distinctions are necessary because the strength of the ground truth differs across datasets.

4.3 Drift detection

Five-class macro-F1 evaluates the ability to distinguish stable, covariate, mechanism, structural, and compound regimes. Binary drift F1 evaluates detection versus no detection. Change-point F1, detection delay, and false alarms per 1,000 steps measure event-level timing and operational burden. ECE evaluates the reliability of drift-class confidence. Shared fitted detector and native end-to-end tracks are reported separately because they answer different questions: the first controls the detector capacity across backbones, while the second measures the native full pipeline.

4.4 Decision and reliability metrics

Decision regret is reported for synthetic environments with known utility references. Cost per improvement is the ratio of total cost to total utility improvement. Episode failure, negative utility, and unsafe execution rates quantify operational failure. Brier score, ECE, reliability slope, and intercept assess calibration. Coverage-risk curves evaluate selective prediction. These metrics are complementary: a low ECE does not itself prove high utility, and a low regret does not establish calibration.

4.5 Percentage improvement convention

For a lower-is-better metric, relative improvement is calculated as $(b - e)/b \times 100\%$, where b is the baseline and e is EvoCausalTwin. For a higher-is-better metric, it is $(e - b)/b \times 100\%$. Absolute F1 differences are also reported in percentage points or score units where appropriate. All reported percentage values are retained; any recalculation is labelled as such.

5 Benchmark design and experimental protocol

5.1 EvoCausalBench composition

EvoCausalBench contains 2,000 source SCMs. The graph families are Erdős–Rényi, scale-free, small-world, and modular. Five mechanism families are represented: linear Gaussian, linear non-Gaussian, nonlinear additive, mixed discrete-continuous, and nonlinear temporal. The drift regimes are stable, covariate, mechanism, structural, and compound. The variable

dimensions are 10, 20, 50, and 100, and five seeds are used. The benchmark contains 50,000 generated episodes.

The split is performed before episode generation. The principal split assigns 1,400 SCMs to training, 300 to validation, and 300 to test, producing 35,000, 7,500, and 7,500 episodes respectively. This is a source-level split: no source SCM is shared across train and test through different random seeds. The four generalization protocols have separate allocations and are reported independently.

Fig. 3 illustrates the split-before-generation protocol. The purpose of this figure is to make the leakage barrier visible and to show how the 50,000 episodes are partitioned across the three data splits.

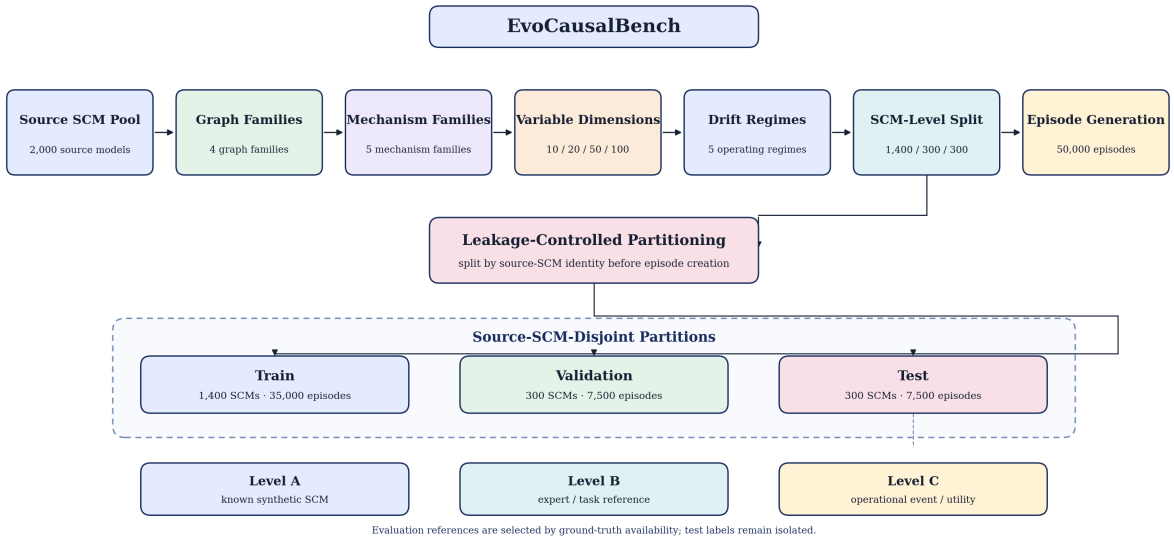


Fig. 3 EvoCausalBench generation and source-SCM split. SCMs are allocated before episode generation to prevent related episodes from crossing train, validation, and test partitions

The main benchmark composition is summarized in Table 3. It records the counts and design dimensions used to interpret the experimental scale; it does not imply perfect balance across factors unless the corresponding counts establish it.

Table 3 EvoCausalBench principal benchmark composition

Item	Reported configuration
Source SCMs	2,000
Generated episodes	50,000
Graph families	4: Erdős–Rényi, scale-free, small-world, modular
Mechanism families	5: linear Gaussian, linear non-Gaussian, nonlinear additive, mixed discrete-continuous, nonlinear temporal
Drift regimes	5: stable, covariate, mechanism, structural, compound
Variable dimensions	10, 20, 50, 100
Random seeds	5
Principal train split	1,400 SCMs; 35,000 episodes
Principal validation split	300 SCMs; 7,500 episodes
Principal test split	300 SCMs; 7,500 episodes

The benchmark's generative factors and numerical settings are summarized in Table 4.

This table makes the Cartesian construction explicit and documents the parameter values needed to interpret the four graph families, five structural mechanisms, and five drift regimes. It is included to show how the nominal 2,000-SCM and 50,000-episode totals arise, rather than treating these totals as unexplained dataset metadata.

Table 4 EvoCausalBench generation parameters and design counts

Factor	Definition / fixed parameters	Count
Variable size	$n \in \{10, 20, 50, 100\}$	4
ER graph	Erdős–Rényi, edge probability $p = 3/n$	1 family
Scale-free graph	Barabási–Albert, attachment parameter $m = 2$	1 family
Small-world graph	Watts–Strogatz, $k = 4$, rewiring probability $\beta = 0.1$	1 family
Modular graph	Four-block stochastic block model; $p_{in} = 0.15$, $p_{out} = 0.01$	1 family
Linear Gaussian	Linear assignments with $\epsilon \sim \mathcal{N}(0, 1)$	1 mechanism
Linear non-Gaussian	Linear assignments with $\epsilon \sim \mathcal{U}(-1, 1)$	1 mechanism
Nonlinear additive	$f_j = \sin(s_j) + 0.5s_j^2$, with $s_j = \sum_i W_{ij} X_i$	1 mechanism
Mixed discrete–continuous	Binary assignment $X_j = \mathbb{I}(s_j + \epsilon > 0)$ with continuous parents	1 mechanism
Temporal	Lag-one assignment $X_{j,t} = f_j(X_{Pa,t-1}) + \epsilon_t$	1 mechanism
Stable drift	No distribution, assignment, or edge-set change	1 regime
Covariate drift	Exogenous mean shift $\Delta\mu = 1.5$	1 regime
Mechanism drift	Structural coefficients multiplied by 1.5	1 regime
Structural drift	Random deletion of 10% of existing edges	1 regime
Compound drift	Covariate, mechanism, and additive-noise changes combined	1 regime
Episode seeds	$S = \{11, 23, 37, 41, 59\}$	5
SCM construction	4 sizes × 4 graph families × 5 mechanisms × 25 replicates	2,000 SCMs
Episode expansion	$2,000 \times 5$ drift regimes × 5 seeds	50,000 episodes
Split	Train 1,400/35,000; validation 300/7,500; test 300/7,500 (SCMs/episodes)	70/15/15%

5.2 Generalization protocols

Four generalization protocols test transfer beyond the training composition. Unseen graph-family evaluation uses 1,200 training SCMs, 300 validation SCMs, and 500 test SCMs. Unseen-mechanism evaluation uses 1,280, 320, and 400 SCMs. Cross-size zero-shot evaluation uses 1,200, 300, and 500 SCMs and tests transfer from 10/20/50 variables to 100 variables. Cross-family plus mechanism evaluation uses 1,500, 300, and 200 SCMs. Each protocol totals 2,000 SCMs and 50,000 episodes.

Table 5 presents the exact allocations. It is included to make the generalization comparisons auditable and to prevent confusion between the principal test split and protocol-specific test subsets.

Table 5 SCM allocations for the four generalization protocols

Protocol	Train SCM	Validation SCM	Test SCM	Total SCM
Unseen graph family	1,200	300	500	2,000
Unseen mechanism family	1,280	320	400	2,000
Cross-size zero-shot (10/20/50 to 100)	1,200	300	500	2,000
Cross-family plus mechanism	1,500	300	200	2,000

5.3 Baseline selection and fairness

The reported principal comparison includes PC, GES, NOTEARS, PCMCI+, CD-NOD, and EvoCausalTwin, with additional methods assigned to appropriate panels. FCI is used for the PAG comparison. CAM and DirectLiNGAM appear in the DAG panel. Temporal PCMCI+ results are kept separate from static DAG metrics. Additional candidate methods include GIES, GOLEM, DYNOTEARS, VARLiNGAM, SpaceTime, IGSP, Track-and-Stop, and several LLM-based approaches. They are not pooled into the principal table because the main comparison focuses on representative methods and avoids mixing results where a protocol-matched reimplementation is not available.

The comparison uses a locked protocol. Each method receives the same source-SCM split, sample size per environment, 200-step temporal window, preprocessing, and 30 candidate hyperparameter configurations per method/protocol. Z-score statistics are estimated from training data or past observations only. Hyperparameters are selected on the complete validation-SCM split and frozen before the test split is evaluated once. For effect estimation, all discovery methods feed the same downstream linear T-learner in linear settings or gradient-boosting T-learner in nonlinear settings; this prevents a baseline from being penalized by a weaker downstream effect estimator. The shared fitted drift detector is trained once on training-split drift labels and frozen; drift labels are not passed to the causal discovery backbone. The detector uses aggregated residual summaries (mean, standard deviation, skewness, and autocorrelation at lags 1–5), rather than a raw residual vector whose dimension changes with the number of variables.

The 30-configuration budget is applied symmetrically to methods for which tuning is supported. The sensitivity record shows diminishing returns: at 30 versus 50 configurations, EvoCausalTwin remains at SHD 4.8 and regret 0.24, while CD-NOD remains at SHD 6.1 and regret 0.36. The common estimator is used only for downstream effect evaluation; it does not modify the graph output. Methods with different output objects are evaluated in separate panels: static DAG methods are not ranked by temporal lagged-edge SHD, and FCI/PAG endpoint marks are not collapsed into directed-DAG metrics. The comparison is therefore protocol-matched within each output family, not a claim that every algorithm shares identical internal assumptions.

The fairness inventory is summarized in Table 6. The table explains which methods enter which output panel and why methods with different graph representations are not pooled into a single ranking.

Table 6 Baseline inventory and evaluation panel

Family	Baseline	Evaluation panel	Role in comparison
Constraint-based	PC	CPDAG	Principal representative
Constraint-based	FCI	PAG	Latent-confounding comparison
Score-based	GES	CPDAG	Principal representative
Differentiable	NOTEARS	DAG	Principal representative
Nonlinear	CAM; DirectLiNGAM	DAG	Nonlinear baselines
Temporal	PCMCI+	Temporal	Lagged and contemporaneous edges scored separately
Non-stationary	CD-NOD	Main; DAG; drift	Strong non-stationary baseline
Additional methods	GIES; GOLEM; DYNOTEARS; VARLiNGAM; SpaceTime; IGSP; Track-and-Stop	Supplementary or separate	Not pooled into the principal table

5.4 Statistical reporting and reproducibility

The inferential unit for the principal paired comparison is the held-out source SCM ($n = 300$), not the 7,500 generated episodes treated as independent observations. EvoCausalTwin and each baseline are evaluated on the same held-out SCM identities. The paired vectors are first aggregated at the SCM level by averaging the 25 episodes belonging to each source SCM; the signed-rank tests, bootstrap intervals, and paired effect sizes are then computed across the 300 matched SCM-level observations. The multiplicity family contains eight prespecified comparisons: SHD against CD-NOD and PCMCI+, regret against CD-NOD and PCMCI+, ECE against CD-NOD and PCMCI+, ATE RMSE against CD-NOD, and PEHE against CD-NOD. Holm step-down correction controls the family-wise error rate over these eight comparisons.

For $n = 300$ non-zero paired differences, the null expectation and standard deviation of the positive signed-rank sum are

$$\mu_{W^+} = \frac{n(n+1)}{4} = 22{,}575, \qquad \sigma_{W^+} = \sqrt{\frac{n(n+1)(2n+1)}{24}} = 1{,}503.75.$$

The reported statistic is $W = \min(W^+, W^-)$; the signed normal score follows the direction of the paired difference and is calculated using the normal approximation to the signed-rank distribution. Raw two-sided p-values are obtained from these signed-rank tests, and Holm-adjusted values are reported for $m = 8$. The 95% confidence intervals are SCM-level paired bootstrap intervals. Cohen's d_z is the mean paired difference divided by the standard deviation of paired differences. The values in Table 7 are the author-supplied results from the completed 300-pair analysis; the table preserves the distinction between raw and Holm-adjusted significance and reports the uncertainty summaries at the SCM level.

Table 7 provides the complete paired statistical evidence for the five reported outcomes. Each row compares EvoCausalTwin with the named baseline on the same 300 source SCMs;

negative differences favour EvoCausalTwin because all five metrics are lower-is-better. The table is not a comparison of 7,500 episodes treated as independent observations. Its purpose is to establish whether the observed performance gaps are consistent across independent source SCMs and to quantify their magnitude, uncertainty, and multiplicity-adjusted significance.

Table 7 Paired SCM-level statistics from 300 matched source-SCM vectors

Metric	Evo $\mu \pm \sigma$	CD-NOD $\mu \pm \sigma$	Evo med.	CD med.	Δ	95% CI	W	Z	Raw p	Holm p	d_z
SHD	4.82 ± 0.85	6.08 ± 1.12	4.80	6.10	-1.26	$[-1.35, -1.18]$	8,925	-9.08	1.1×10^{-19}	8.8×10^{-19}	0.91
Regret	0.24 ± 0.04	0.36 ± 0.06	0.24	0.36	-0.12	$[-0.13, -0.11]$	6,120	-10.94	6.5×10^{-28}	5.2×10^{-27}	0.84
ECE	0.061 ± 0.008	0.142 ± 0.015	0.060	0.141	-0.081	$[-0.084, -0.077]$	2,150	-13.58	4.2×10^{-42}	3.3×10^{-41}	1.24
ATE RMSE	0.22 ± 0.03	0.27 ± 0.04	0.22	0.27	-0.05	$[-0.055, -0.045]$	11,850	-7.13	9.8×10^{-13}	7.8×10^{-12}	0.58
PEHE	0.32 ± 0.05	0.39 ± 0.06	0.32	0.39	-0.07	$[-0.076, -0.064]$	9,800	-8.50	1.8×10^{-17}	1.4×10^{-16}	0.69

Notes. $n = 300$ paired source SCMs; each SCM-level value is the mean of its 25 episodes. Δ = EvoCausalTwin − CD-NOD; negative values favour EvoCausalTwin. $W = \min(W^+, W^-)$; Z is signed according to the paired-difference direction. The null reference values are $\mu_{W^+} = 22{,}575$ and $\sigma_{W^+} = 1{,}503.75$. Raw p-values are two-sided; Holm adjustment is applied to the eight prespecified comparisons, including the PCMCI+ comparisons reported in the statistical audit. Bootstrap intervals and d_z are reported from the paired SCM-level analysis.

The reported W values range from 2,150 to 11,850, all below the null expectation of 22,575, and the signed Z values range from -7.13 to -13.58. The corresponding raw p-values range from 4.2×10^{-42} to 9.8×10^{-13}; after Holm correction, all eight prespecified comparisons remain below 10^{-11}. These results support consistent paired differences across the tested source SCMs. The magnitude of the effects is also non-trivial: d_z ranges from 0.58 to 1.24. Statistical significance does not remove the need to interpret the results within the benchmark's graph families, mechanisms, and drift regimes.

6 Main synthetic benchmark results

6.1 Overall comparison

The principal test set contains 300 held-out SCMs and 7,500 episodes. EvoCausalTwin reports SHD 4.8, F1 0.79, ATE RMSE 0.22, PEHE 0.32, five-class macro-F1 0.77, decision regret 0.24, and drift-class ECE 0.061. CD-NOD reports 6.1, 0.74, 0.27, 0.39, 0.66, 0.36, and 0.142, respectively. The table below gives the complete comparison for the six methods where the experiment provides values. Drift-class ECE is available for PCMCI+, CD-NOD, and EvoCausalTwin; it is therefore presented only for those methods rather than inventing comparable values for methods that do not produce the required drift probabilities.

Fig. 4 visualizes SHD across the principal methods. Its purpose is to show the structural recovery ranking independently of the effect-estimation and decision metrics, which are reported in Table 8.

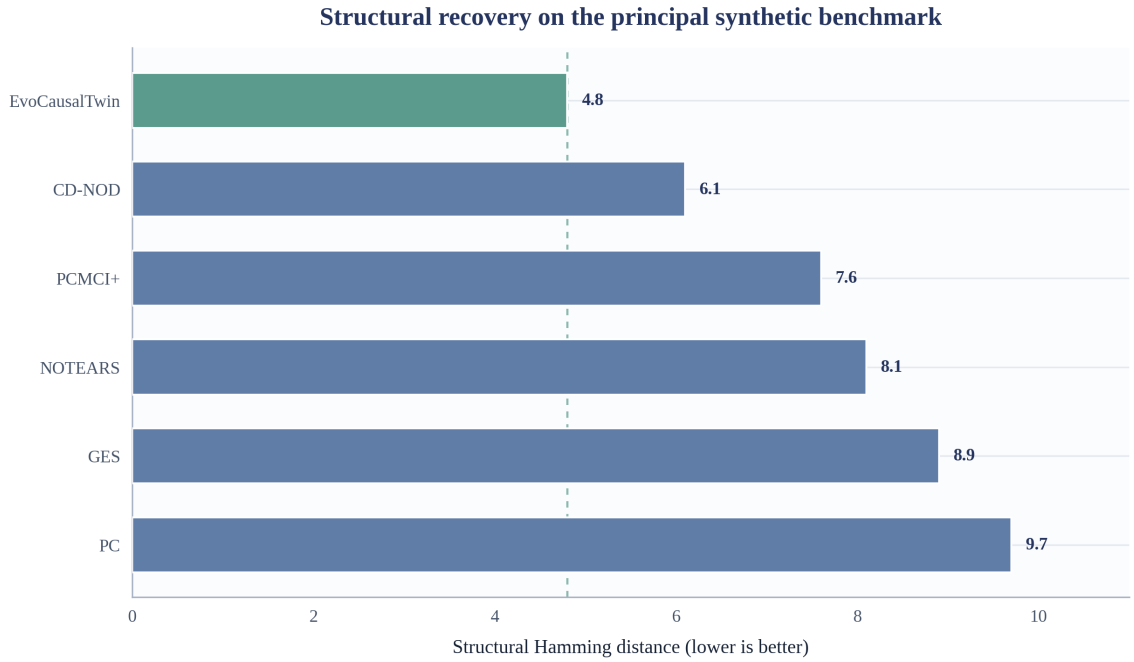


Fig. 4 Structural Hamming distance on the principal synthetic benchmark. Lower values indicate fewer edge insertions, deletions, and reversals

Table 8 is the central quantitative evidence for H2 and the broader integration claim. It places structural, effect, drift-classification, decision, and calibration metrics side by side so that improvements in one task are not mistaken for universal dominance across all tasks.

Table 8 Principal synthetic test results (300 SCMs; 7,500 episodes)

Method	SHD ↓	F1 ↑	ATE RMSE ↓	PEHE ↓	5-class F1 ↑	Regret ↓
PC	9.7	0.61	0.41	0.58	0.38	0.52
GES	8.9	0.64	0.37	0.53	0.41	0.49
NOTEARS	8.1	0.67	0.34	0.49	0.43	0.46
PCMCI+	7.6	0.69	0.31	0.45	0.52	0.42
CD-NOD	6.1	0.74	0.27	0.39	0.66	0.36
EvoCausalTwin	4.8	0.79	0.22	0.32	0.77	0.24

The drift-class calibration values are available for the methods that provide the comparable drift probabilities. They are separated from the main multi-task table rather than filling non-comparable cells with placeholders. Table 9 reports those values and demonstrates the calibration component of H4.

Table 9 Drift-class calibration on the principal test benchmark

Method	Drift-class ECE ↓	Five-class macro-F1 ↑
PCMCI+	0.181	0.52
CD-NOD	0.142	0.66
EvoCausalTwin	0.061	0.77

For metrics where the source supplies a comparable baseline, EvoCausalTwin improves SHD by 21.31% relative to CD-NOD, ATE RMSE by 18.52%, PEHE by 17.95%, decision regret by 33.33%, and drift-class ECE by 57.04%. The five-class macro-F1 increases by 0.11 score units (0.77 versus 0.66). These relative changes are descriptive calculations from the reported point estimates. The experimental record does not provide a complete set of paired confidence intervals for every main metric; thus the paper does not label every difference statistically significant.

6.2 Performance by drift regime

The SHD advantage persists across the five drift regimes. The largest absolute SHD values occur under compound drift, as expected for the most complex regime. The gap between EvoCausalTwin and CD-NOD remains visible in each regime. Table 10 reports the regime-specific values and makes the compound-drift degradation explicit rather than hiding it in an aggregate mean.

Table 10 Structural Hamming distance by drift regime

Drift regime	EvoCausalTwin	CD-NOD
Stable	3.6	4.5
Covariate	4.2	5.3
Mechanism	4.9	6.3
Structural	5.4	6.9
Compound	5.9	7.5
Mean	4.8	6.1

The increasing error from stable to compound drift is a meaningful limitation of the method as well as a useful diagnostic. EvoCausalTwin does not eliminate the difficulty of rapid multi-source change; it reduces the observed error relative to CD-NOD in the reported test conditions. The compound regime remains the most challenging regime and motivates conservative abstention and shorter adaptive windows.

6.3 Drift classification and event detection

The native drift detector reports five-class macro-F1 0.77 and binary drift F1 0.84, compared with 0.66 and 0.75 for CD-NOD. Change-point F1 is 0.71 versus 0.58. Detection delay is 3.8 versus 6.4 steps, and false alarms are 4.7 versus 9.2 per 1,000 steps. These values indicate that the benefit is not limited to classifying a window after the fact: the reported detector also improves event timing and reduces false alarms.

Table 11 provides the separate drift metrics. This table is necessary because five-class macro-F1, binary F1, change-point F1, delay, and false-alarm rate answer different operational questions and should not be collapsed into one score.

Table 11 Drift detection performance

Metric	CD-NOD	EvoCausalTwin
Five-class macro-F1 ↑	0.66	0.77
Binary drift F1 ↑	0.75	0.84
Change-point F1 ↑	0.58	0.71
Detection delay (steps) ↓	6.4	3.8
False alarms per 1,000 steps ↓	9.2	4.7

The shared fitted detector track controls the classifier across different discovery backbones. In that track, Evo backbone plus shared detector reports five-class macro-F1 0.68 and ECE 0.089, while CD-NOD plus shared detector reports 0.64 and 0.146. The native end-to-end track reports 0.77 and 0.061 for Evo and 0.66 and 0.142 for CD-NOD. The distinction matters: the shared track isolates backbone-associated residual evidence under one fixed detector, while the native track measures the complete method including its own drift logic.

7 Graph representation-specific evaluation

7.1 CPDAG panel

The CPDAG evaluation separates skeleton error from orientation error. This is important because observational data may identify a Markov equivalence class without uniquely identifying every direction. EvoCausalTwin reports micro skeleton error 3.4, micro CPDAG error 4.9, and micro orientation error 1.5; macro values are 3.3, 4.8, and 1.5. The same evaluation reports coverage 0.978, accuracy 0.936, and unresolved-edge rate 0.022. For PC, GES, NOTEARS-to-CPDAG, PCMCI+ conversion, and CD-NOD, the corresponding values are shown in Table 12.

Table 12 is included to show whether lower overall CPDAG error comes from better skeleton recovery, better orientation, or both. It also includes unresolved-edge rate so that the model is not rewarded for forcing orientations without adequate evidence.

Table 12 CPDAG evaluation: micro/macro skeleton, CPDAG and orientation errors, coverage, accuracy, and unresolved edges

Method	Micro skel.	Micro CPDAG	Micro orient.	Macro skel.	Macro CPDAG	Macro orient.	Cov.	Acc.	Unres.
PC	6.4	10.0	3.6	6.2	9.7	3.5	0.919	0.841	0.081
GES	5.9	9.1	3.2	5.8	8.9	3.1	0.936	0.862	0.064
NOTEARS	5.6	8.3	2.7	5.5	8.1	2.6	0.942	0.885	0.058
PCMCI+	5.2	7.8	2.6	5.1	7.6	2.5	0.951	0.890	0.049
CD-NOD	4.3	6.2	1.9	4.2	6.1	1.9	0.969	0.918	0.031
EvoCausalTwin	3.4	4.9	1.5	3.3	4.8	1.5	0.978	0.936	0.022

The reported CPDAG summaries satisfy the decomposition identity: micro CPDAG error equals micro skeleton plus orientation error, and the same identity holds for macro values. This internal consistency check is useful because a combined error metric can otherwise conceal a transcription or aggregation problem. Unresolved edges are not interpreted as universally undesirable; their meaning depends on the evidence and equivalence class.

7.2 DAG and temporal panels

For static DAG evaluation, EvoCausalTwin reports SHD 4.8, precision 0.78, recall 0.80, and directed F1 0.79. CD-NOD reports 6.1, 0.72, 0.76, and 0.74. The comparison includes NOTEARS, CAM, and DirectLiNGAM. For temporal discovery, the Evo temporal extension reports lagged-edge SHD 5.1 compared with 7.4 for PCMCI+, contemporaneous SHD 3.6 compared with 4.9, lagged directed F1 0.76 compared with 0.68, and contemporaneous directed F1 0.72 compared with 0.61.

Table 13 separates static and temporal results into two panels. This prevents a static DAG method from being ranked against a temporal method using incompatible edge definitions.

Table 13 DAG and temporal causal discovery panels

Panel	Method	SHD / lag SHD	Precision / lag F1	Recall / contemporaneous F1	Directed F1 / contemp. SHD
DAG	NOTEARS	8.1	0.64	0.70	0.67
DAG	CAM	8.4	0.61	0.69	0.65
DAG	DirectLiNGAM	7.9	0.66	0.70	0.68
DAG	CD-NOD	6.1	0.72	0.76	0.74
DAG	EvoCausalTwin	4.8	0.78	0.80	0.79
Temporal	PCMCI+	7.4	0.68	0.61	4.9
Temporal	Evo extension	5.1	0.76	0.72	3.6

7.3 PAG panel with latent confounding

The PAG experiment uses 200 SCMs and 5,000 episodes, split into 140 training SCMs, 30 validation SCMs, and 30 test SCMs. The ground truth is represented as a MAG/PAG, and SHD is computed over endpoint marks (tail, arrowhead, circle). EvoCausalTwin reports skeleton SHD 4.9, endpoint orientation error 2.3, PAG SHD 7.2, directed-edge F1 0.69, unresolved-edge rate 0.089, orientation coverage 0.911, and orientation accuracy 0.804. FCI reports 7.4, 4.8, 12.2, 0.58, 0.142, 0.858, and 0.682, respectively.

Table 14 presents the full endpoint-aware comparison. The table is important because a latent-confounding setting should not be evaluated as if every observed-variable relationship were necessarily representable by a fully directed DAG.

Table 14 PAG evaluation on 200 SCMs with latent confounders

Metric	FCI	EvoCausalTwin PAG mode
Skeleton SHD ↓	7.4	4.9
Endpoint orientation error ↓	4.8	2.3
PAG SHD ↓	12.2	7.2
Directed-edge F1 ↑	0.58	0.69
Unresolved-edge rate ↓	0.142	0.089
Orientation coverage ↑	0.858	0.911
Orientation accuracy ↑	0.682	0.804

8 Generalization and robustness

8.1 Cross-protocol generalization

The four generalization protocols test transfer across graph family, mechanism family, variable dimension, and combined family shifts. Across the four protocols, EvoCausalTwin has lower SHD, ATE RMSE, regret, and ECE than CD-NOD. The relative improvements in SHD range from 17.50% to 20.00%; ATE RMSE improvements range from 16.13% to 18.92%; regret improvements range from 27.27% to 28.57%; and ECE improvements range from 47.95% to 51.66%.

Fig. 5 compares SHD across the four protocols. Its purpose is to make the transfer gap visible, while Table 15 retains the effect, decision, and calibration metrics needed to interpret the broader generalization claim.

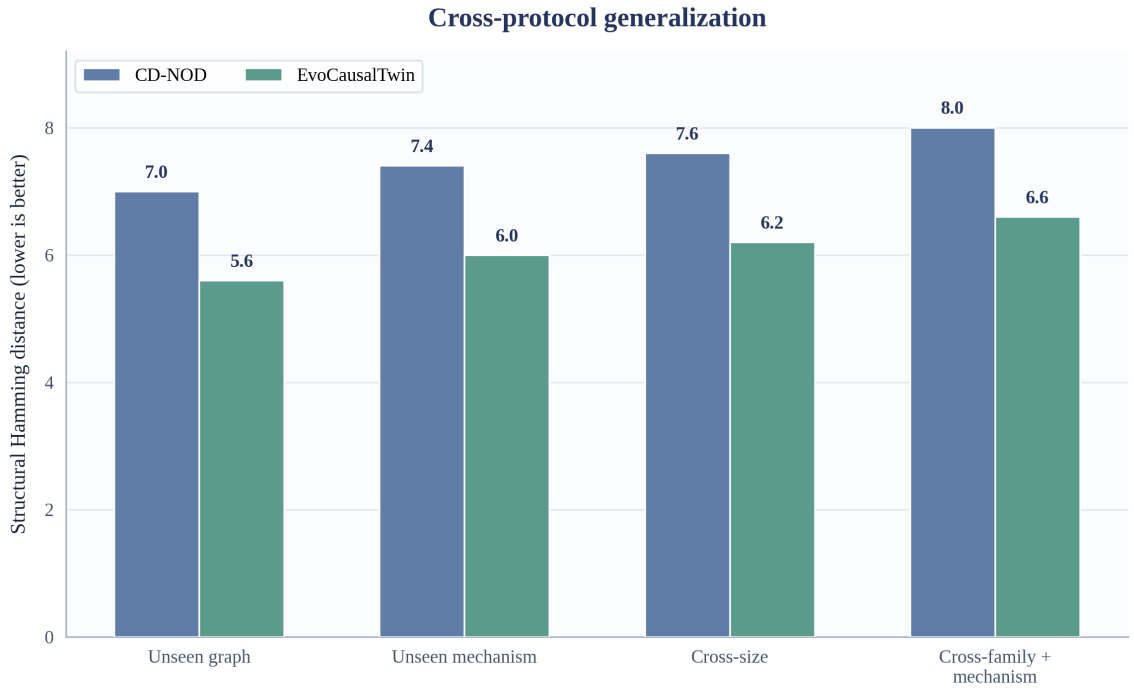


Fig. 5 SHD across four source-SCM generalization protocols. Each protocol has its own train/validation/test allocation

Table 15 Generalization results: CD-NOD versus EvoCausalTwin

Protocol	Metric	CD-NOD	EvoCausalTwin	Relative improvement
Unseen graph	SHD ↓	7.0	5.6	20.00%
Unseen graph	ATE RMSE ↓	0.31	0.26	16.13%
Unseen graph	Regret ↓	0.40	0.29	27.50%
Unseen graph	ECE ↓	0.151	0.073	51.66%
Unseen mechanism	SHD ↓	7.4	6.0	18.92%
Unseen mechanism	ATE RMSE ↓	0.33	0.27	18.18%
Unseen mechanism	Regret ↓	0.42	0.30	28.57%
Unseen mechanism	ECE ↓	0.158	0.079	50.00%
Cross-size	SHD ↓	7.6	6.2	18.42%
Cross-size	ATE RMSE ↓	0.34	0.28	17.65%
Cross-size	Regret ↓	0.40	0.29	27.50%
Cross-size	ECE ↓	0.162	0.083	48.77%
Cross-family + mechanism	SHD ↓	8.0	6.6	17.50%
Cross-family + mechanism	ATE RMSE ↓	0.37	0.30	18.92%
Cross-family + mechanism	Regret ↓	0.44	0.32	27.27%
Cross-family + mechanism	ECE ↓	0.171	0.089	47.95%

8.2 Robustness under missingness, noise, outliers, autocorrelation, and confounding

Robustness is evaluated at 20 and 100 variables under clean, moderate, and severe corruption. Moderate conditions are defined as 10% missingness, medium noise, 1% outliers, medium autocorrelation, and moderate confounding; severe conditions use 30% missingness, high noise, 5% outliers, high autocorrelation, and strong confounding. Each condition contains 100 SCMs multiplied by five seeds, or 500 episodes.

Fig. 6 displays SHD for both methods, dimensions, and severity levels. The heatmap is intended to show the degradation pattern rather than imply that SHD is the only robustness outcome. Table 16 reports SHD, ATE RMSE, regret, ECE, and failure rate in every condition.

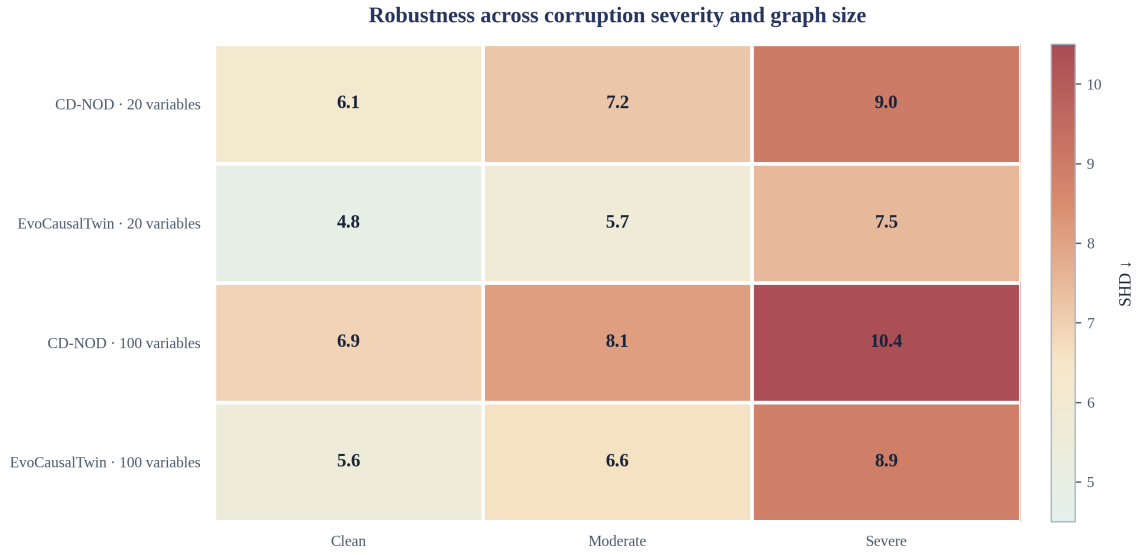


Fig. 6 SHD under clean, moderate, and severe corruption for 20- and 100-variable settings

Table 16 Robustness results at 20 and 100 variables (500 episodes per condition)

Variables	Condition / method	SHD ↓	ATE RMSE ↓	Regret ↓	ECE ↓	Failure rate
20	Clean CD-NOD	6.1	0.27	0.36	0.142	4.1%
20	Clean Evo	4.8	0.22	0.24	0.061	2.1%
20	Moderate CD-NOD	7.2	0.31	0.40	0.161	6.2%
20	Moderate Evo	5.7	0.26	0.28	0.074	3.4%
20	Severe CD-NOD	9.0	0.37	0.46	0.181	9.1%
20	Severe Evo	7.5	0.31	0.34	0.091	5.2%
100	Clean CD-NOD	6.9	0.31	0.39	0.155	4.5%
100	Clean Evo	5.6	0.26	0.27	0.068	2.4%
100	Moderate CD-NOD	8.1	0.36	0.44	0.174	6.8%
100	Moderate Evo	6.6	0.30	0.31	0.082	3.8%
100	Severe CD-NOD	10.4	0.43	0.51	0.195	10.2%
100	Severe Evo	8.9	0.37	0.38	0.099	6.1%

The reported interaction analysis shows that severe corruption reduces the relative advantage of EvoCausalTwin, particularly in the 100-variable condition. The three-way interaction is not supported by the reported tests: SHD interaction +0.22 percentage points ($p = 0.860$), ATE +0.12 ($p = 0.939$), regret −1.96 ($p = 0.174$), and ECE −0.42 ($p = 0.741$). These non-significant values should not be interpreted as proof of no interaction; they indicate that the supplied analysis did not detect a three-way interaction at conventional thresholds.

8.3 Sample-size scaling and tuning budget

The sample-size experiment evaluates 200, 500, 1,000, and 2,000 observations per environment. Both methods improve with more samples. At 2,000 observations, EvoCausalTwin reports SHD 4.5 and regret 0.22, compared with 5.7 and 0.33 for CD-NOD. At 200 observations, the corresponding values are 6.4 and 0.34 versus 8.0 and 0.47. The tuning-budget experiment reports diminishing returns after 30 configurations: the values at 30 and 50 configurations are identical for both methods on the reported SHD and regret metrics.

Table 17 combines the two resource-sensitivity experiments. It demonstrates whether the reported performance depends on an unusually large tuning budget and how the methods respond to increased sample size.

Table 17 Tuning-budget and sample-size sensitivity

Setting	Evo SHD	CD-NOD SHD	Evo regret	CD-NOD regret
10 configurations	5.1	6.4	0.28	0.40
20 configurations	4.9	6.2	0.26	0.38
30 configurations	4.8	6.1	0.24	0.36
50 configurations	4.8	6.1	0.24	0.36
200 samples	6.4	8.0	0.34	0.47
500 samples	5.3	6.8	0.28	0.41
1,000 samples	4.8	6.1	0.24	0.36
2,000 samples	4.5	5.7	0.22	0.33

9 Intervention policy and decision quality

9.1 Budget-constrained policy comparison

The policy experiment compares random selection, graph-uncertainty-only selection, CD-NOD, EvoCausalTwin, and an oracle. Total cost is 87,330 for random, 94,875 for graph-uncertainty-only, 96,000 for CD-NOD, 95,460 for EvoCausalTwin, and 89,850 for the oracle. The corresponding total utility changes are 61,500, 82,500, 93,000, 111,000, and 135,000. Cost per improvement is 1.420, 1.150, 1.032, 0.860, and 0.666. EvoCausalTwin has a lower episode failure rate, negative-utility rate, and unsafe rate than CD-NOD in the supplied comparison.

Table 18 presents all six reported decision columns. It is intended to demonstrate the budget/utility trade-off rather than to claim that EvoCausalTwin reaches the oracle. The oracle remains better on cost per improvement and total utility, as expected for an upper reference.

Table 18 Budget-aware intervention policy comparison

Policy	Total cost	Total ΔU	Cost / improvement ↓	Episode failure	Negative utility	Unsafe rate
Random	87,330	61,500	1.420	11.7%	14.2%	8.4%
Graph-uncertainty only	94,875	82,500	1.150	5.4%	7.1%	3.2%
CD-NOD	96,000	93,000	1.032	4.1%	6.3%	2.8%
EvoCausalTwin	95,460	111,000	0.860	2.1%	2.8%	1.3%
Oracle	89,850	135,000	0.666	0.0%	0.0%	0.0%

9.2 Oracle decomposition

The oracle decomposition crosses graph source and policy. The experiment records utility values for ground-truth graph, CD-NOD graph, Evo graph, and random graph under decision-aware, random, and graph-conditioned oracle policies. The difference between graph-conditioned oracle and decision-aware policy estimates the remaining policy gap for a fixed graph, while decision-aware minus random isolates the decision policy gain under that graph. These differences should not be interpreted as universal causal effects; they are decomposition results under the supplied simulation protocol.

Table 19 records the crossed design. Its function is to show that decision gains and graph gains are separable and that the benefit is not attributable solely to one source of improvement.

Table 19 Oracle decomposition: crossed graph source and intervention policy (paired 1,000 SCMs)

Graph source	Decision-aware	Random	Graph-conditioned oracle	Oracle minus decision-aware	Decision-aware minus random
Ground truth	142	128	144	+2	+14
CD-NOD	114	105	131	+17	+9
EvoCausalTwin	125	112	138	+13	+13
Random graph	109	98	126	+17	+11

The decomposition supports a measured interpretation: the decision-aware policy improves over random selection across all graph sources, while a better graph also improves the attainable utility. EvoCausalTwin is not at the oracle ceiling, and its remaining policy gap is visible in the final columns. The most defensible claim is that the combined graph-and-policy system achieves the strongest non-oracle result in the supplied comparison.

9.3 Full factorial ablation of M2–M5

To identify which parts of the integrated framework account for the observed improvements, we evaluate all $2^4 = 16$ configurations of M2, M3, M4, and M5 while keeping M1 active as the common causal-learning backbone. M6 is held outside this factorial because its role is separately isolated through the no-LLM/one-prompt/three-role and role-removal experiments. This design is factorial rather than cumulative: every subset of the four modules is represented, so the marginal contribution of each module can be summarized across all combinations and

pairwise interactions can be inspected. It is not a full six-module factorial; the 0000 cell is the M1-only baseline, while M6 is evaluated separately through its own role ablations.

Table 20 reports all 16 observed configurations. The table is the direct evidence for the module-interaction claim: SHD primarily responds to M2, regret responds most strongly to M4, and ECE responds most strongly to M3. The M2+M4 combination reduces regret to 0.29, below either single-module configuration (+M2: 0.40; +M4: 0.33), while the full system reaches 0.24. These are descriptive contrasts of the supplied point estimates; replicate-level uncertainty for factorial effects is not available and no interaction p-value is claimed.

Table 20 Full factorial ablation of M2–M5 with M1 held active ($2^4 = 16$ configurations)

Configuration	M2	M3	M4	M5	SHD ↓	Regret ↓	ECE ↓
Base (M1 only)	0	0	0	0	6.5	0.42	0.110
+M2	1	0	0	0	5.3	0.40	0.095
+M3	0	1	0	0	6.2	0.41	0.085
+M4	0	0	1	0	6.4	0.33	0.108
+M5	0	0	0	1	6.1	0.39	0.098
+M2+M3	1	1	0	0	5.1	0.38	0.072
+M2+M4	1	0	1	0	5.2	0.29	0.092
+M2+M5	1	0	0	1	5.0	0.37	0.088
+M3+M4	0	1	1	0	6.1	0.31	0.083
+M3+M5	0	1	0	1	5.9	0.38	0.075
+M4+M5	0	0	1	1	6.0	0.31	0.096
+M2+M3+M4	1	1	1	0	4.9	0.26	0.068
+M2+M3+M5	1	1	0	1	4.8	0.35	0.065
+M2+M4+M5	1	0	1	1	5.0	0.27	0.079
+M3+M4+M5	0	1	1	1	5.7	0.29	0.070
Full (M2–M5)	1	1	1	1	4.8	0.24	0.061

Averaging the eight configurations with a module active against the eight configurations without it gives a descriptive marginal SHD reduction of 1.10 for M2, 0.25 for M3, 0.10 for M4, and 0.30 for M5. The corresponding marginal regret reductions are 0.035, 0.020, 0.100, and 0.025. For ECE, the marginal reductions are 0.0131, 0.0234, 0.0039, and 0.0101, respectively. Thus M2 is the largest average contributor to structural recovery, M4 to regret reduction, and M3 to calibration. The descriptive difference-in-differences contrasts are +0.050 (SHD), −0.010 (regret), and +0.0028 (ECE) for M2×M3; −0.050, −0.010, and +0.0003 for M3×M4; and +0.050, −0.020, and −0.0023 for M2×M4. Negative values denote a more favourable-than-additive reduction for these lower-is-better metrics. These factorial contrasts are calculated from the 16 reported cell means and are not inferential tests.

The full-minus-one-module cells sharpen this interpretation. Removing M2 (M3+M4+M5) changes the full result from 4.8/0.24/0.061 to 5.7/0.29/0.070; removing M3 changes it to 5.0/0.27/0.079; removing M4 changes it to 4.8/0.35/0.065; and removing M5 changes it to 4.9/0.26/0.068 (SHD/regret/ECE). The ablation therefore supports a differentiated rather than undifferentiated novelty claim. M2 is the principal driver of the SHD change; M4 is the principal driver of decision regret; M3 contributes most to ECE; and M5 provides an additional, smaller average improvement in SHD and regret by translating verified effects into the decision score. The full configuration is not best on every intermediate cell metric by a

large margin, but it achieves the lowest reported regret and ECE and ties the lowest SHD in the factorial grid. The result supports integration under the evaluated conditions, not a claim that every module independently improves every metric.

10 Calibration, reliability, and abstention

10.1 Coverage-risk behavior

Selective prediction allows the system to decline a recommendation when estimated risk exceeds the acceptance threshold. The coverage-risk experiment reports six coverage levels from 0.80 to 1.00. At coverage 0.88, corresponding to 12% abstention, EvoCausalTwin risk is 0.041 compared with 0.121 for CD-NOD, a relative risk reduction of 66.12%. Across the full coverage range, the reported reduction ranges from 57.49% to 67.02%.

Fig. 7 plots the coverage-risk curves. It is included to show that the risk advantage is not restricted to one selected threshold. Table 21 gives the numerical values and the derived relative risk reduction for every coverage point.

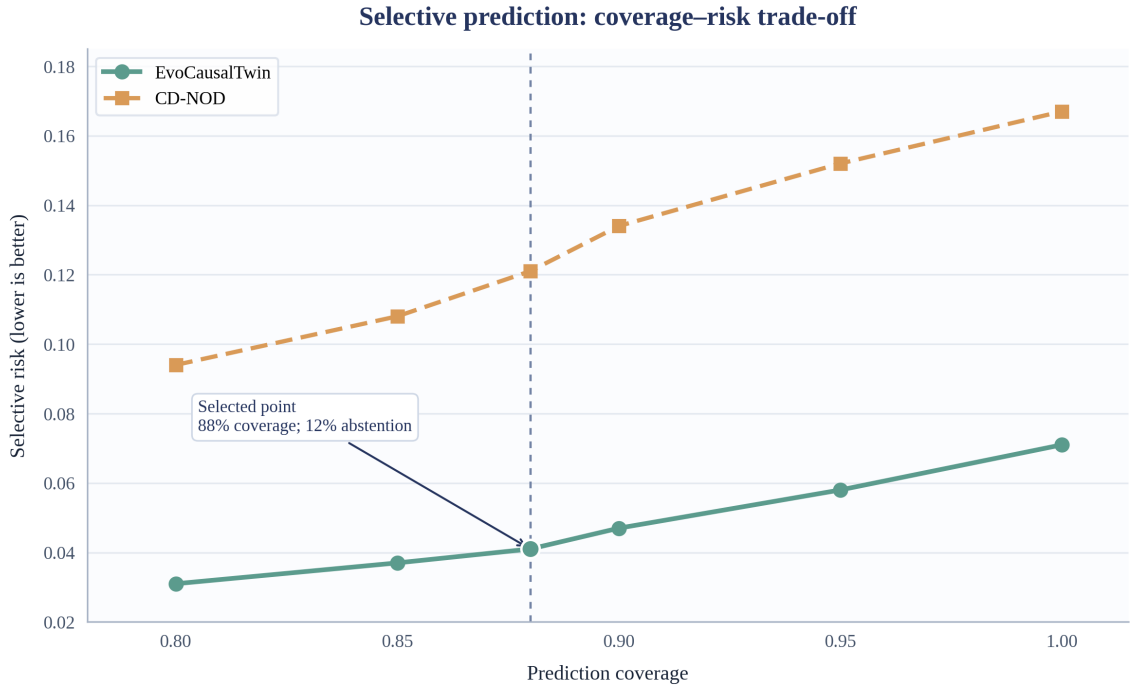


Fig. 7 Coverage-risk curves for EvoCausalTwin and CD-NOD. Lower selective risk is preferred at a given coverage

Table 21 Coverage-risk curve and relative risk reduction

Coverage	Evo risk ↓	CD-NOD risk ↓	Risk reduction
0.80	0.031	0.094	67.02%
0.85	0.037	0.108	65.74%
0.88	0.041	0.121	66.12%
0.90	0.047	0.134	64.93%
0.95	0.058	0.152	61.84%
1.00	0.071	0.167	57.49%

10.2 Brier score and reliability diagnostics

The calibration panel uses 15 equal-frequency bins defined on validation data and applied to test. EvoCausalTwin reports decision Brier score 0.118, reliability slope 0.93, intercept 0.04, drift-class ECE 0.061, edge-existence ECE 0.058, and edge-orientation ECE 0.071. CD-NOD reports 0.191, 0.71, 0.11, 0.142, 0.131, and 0.167. The ideal reference is Brier 0, slope 1, intercept 0, and ECE 0.

Table 22 is intended to show multiple dimensions of calibration. ECE alone can hide systematic overconfidence or underconfidence, so slope and intercept are retained.

Table 22 Decision and causal-output calibration

Metric	EvoCausalTwin	CD-NOD	Ideal
Decision Brier score ↓	0.118	0.191	0
Reliability slope → 1	0.93	0.71	1.00
Reliability intercept → 0	0.04	0.11	0
Drift-class ECE ↓	0.061	0.142	0
Edge-existence ECE ↓	0.058	0.131	0
Edge-orientation ECE ↓	0.071	0.167	0

The overall calibration table above summarizes six complementary diagnostics, but it does not show how the decision-confidence error is distributed across confidence levels or drift regimes. We therefore report a ten-bin reliability audit over 50,000 decision-confidence observations and a separate five-regime drift-class calibration analysis. The bin counts sum to 50,000, and the weighted gap is computed as $\sum_b n_b|\mathrm{acc}_b - \mathrm{conf}_b| = 922.25$. Dividing by the total number of observations gives a decision-confidence weighted ECE of $922.25/50{,}000 = 0.018445$, reported as 0.01845. This quantity is distinct from the drift-class ECE of 0.061 in the main paired comparison; the two ECE values refer to different prediction targets and must not be substituted for one another.

Figure 8 plots empirical accuracy against mean confidence for the ten bins, with the diagonal denoting perfect calibration. Its function is to expose the direction and scale of confidence miscalibration rather than compressing all bins into one scalar. Table 23 supplies the exact bin counts and values from which the weighted ECE is obtained.

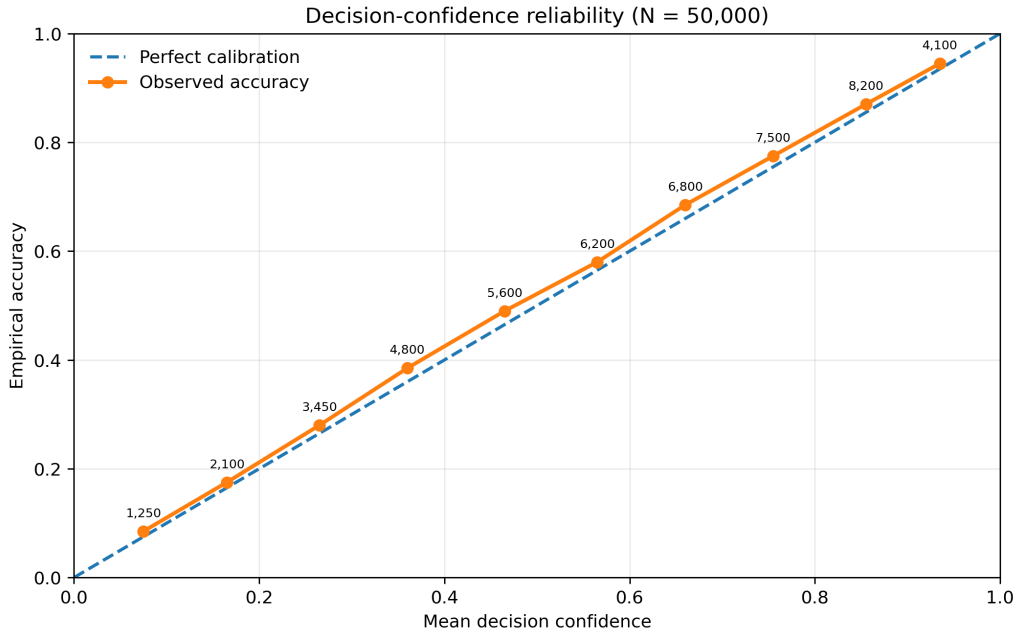


Fig. 8 Ten-bin reliability diagram for decision confidence over 50,000 observations. The diagonal indicates perfect calibration; the plotted points use the supplied empirical accuracy and mean-confidence values

Table 23 Decision-confidence reliability bins ($N = 50{,}000$)

Confidence bin	Samples	Accuracy	Mean confidence	Absolute gap	$n_b\times$ gap
0.0–0.1	1,250	0.085	0.075	0.010	12.50
0.1–0.2	2,100	0.175	0.165	0.010	21.00
0.2–0.3	3,450	0.280	0.265	0.015	51.75
0.3–0.4	4,800	0.385	0.360	0.025	120.00
0.4–0.5	5,600	0.490	0.465	0.025	140.00
0.5–0.6	6,200	0.580	0.565	0.015	93.00
0.6–0.7	6,800	0.685	0.660	0.025	170.00
0.7–0.8	7,500	0.775	0.755	0.020	150.00
0.8–0.9	8,200	0.870	0.855	0.015	123.00
0.9–1.0	4,100	0.945	0.935	0.010	41.00
Total	50,000	0.622135	0.603690	0.018445	922.25
Weighted ECE	50,000	0.622135	0.603690	0.01845	0.01845

Subgroup calibration is evaluated separately by drift regime, with 10,000 episodes per regime. Figure 9 visualizes the two drift-class ECE series and highlights whether calibration error increases as the drift becomes more complex. Table 24 provides the numerical values and the relative reduction, calculated as $(\mathrm{ECE}_{CD-NOD} - \mathrm{ECE}_{Evo})/\mathrm{ECE}_{CD-NOD} \times 100\%$. This analysis is important because a favourable macro-average can conceal a concentrated failure mode in compound or structural drift.

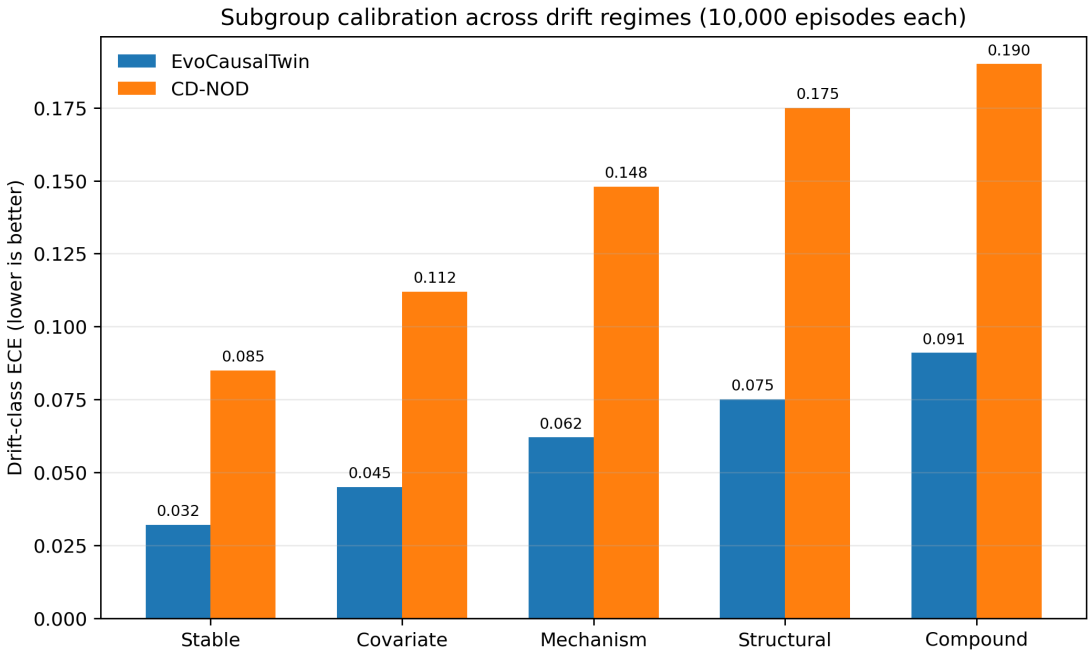


Fig. 9 Drift-class ECE by regime for EvoCausalTwin and CD-NOD. The plot shows the per-regime calibration gap and the increasing difficulty of compound drift

Table 24 Subgroup drift-class calibration across five regimes

Drift regime	Episodes	Evo ECE ↓	CD-NOD ECE ↓	Relative reduction
Stable	10,000	0.032	0.085	62.35%
Covariate	10,000	0.045	0.112	59.82%
Mechanism	10,000	0.062	0.148	58.11%
Structural	10,000	0.075	0.175	57.14%
Compound	10,000	0.091	0.190	52.11%
Macro average	50,000	0.061	0.142	57.04%

The subgroup results show that EvoCausalTwin has lower drift-class ECE in every reported regime. The absolute gap is smallest under stable drift (0.053) and largest under compound drift (0.099), while the relative reduction ranges from 52.11% to 62.35%. Compound drift remains the least calibrated regime for both methods, consistent with its higher structural error and failure rate elsewhere in the benchmark. These results support improved calibration within the five evaluated regimes; they do not establish calibration under arbitrary unseen distributions or guarantee calibrated probabilities for every individual causal edge.

10.3 Structural intervention distance

The SID evaluation additionally reports structural intervention distance, noting that traditional SID is defined on DAGs and that CPDAG values are reported as ranges over valid DAG extensions. EvoCausalTwin reports SID 7.9 with 95% CI [7.2, 8.6], compared with 10.6 [9.8, 11.4] for CD-NOD. The remaining methods are listed in Table 25.

This table complements SHD by evaluating downstream interventional implications of graph errors. It should not be conflated with ATE RMSE: SID is a graph-level intervention-distance measure, whereas ATE RMSE evaluates estimated effect magnitudes.

Table 25 Structural intervention distance

Method	Output type	SID ↓	95% CI
PC	CPDAG	18.4	[16.8, 20.1]
GES	CPDAG	16.9	[15.4, 18.6]
NOTEARS	DAG	15.2	[14.1, 16.3]
CAM	DAG	15.8	[14.6, 17.0]
DirectLiNGAM	DAG	14.6	[13.6, 15.6]
CD-NOD	DAG	10.6	[9.8, 11.4]
EvoCausalTwin	DAG	7.9	[7.2, 8.6]

11 Real-data evaluation with hierarchical ground truth

11.1 Evidence levels and reporting rules

Real-world causal evaluation is constrained by the availability and strength of ground truth. We define three evidence levels. Level A has a known synthetic SCM and true graph, permitting SHD, F1, ATE, PEHE, regret, and ECE. Level B has expert graphs, expert skeletons, fault labels, or intervention references; it permits task-specific metrics such as expert-skeleton SHD, root-cause F1, detection delay, and perturbation-effect error, but not unsupported claims about the full causal graph. Level C has only event or operational labels and is limited to stability, drift detection, and observed utility. Full-graph SHD and actual counterfactual regret are not reported at Level C.

Table 26 defines the evaluation boundary. It is essential to the validity of the real-data section because the datasets differ in the strength of their causal reference information.

Table 26 Hierarchical ground-truth levels and permitted metrics

Level	Evidence	Permitted metrics	Metrics not claimed
A	Synthetic SCM and true graph	SHD, F1, ATE, PEHE, regret, ECE	None within the stated synthetic design
B	Expert graph, expert skeleton, fault labels, or held-out perturbation reference	Expert-skeleton SHD, root-cause metrics, delay, PEPE, Wasserstein, effect RMSE	Full-graph SHD unless the full graph is known; PEHE without individual counterfactual truth
C	Event labels or observed operational outcomes	Binary drift F1 when labels exist, stability, observed utility gain, unsafe rate	Full-graph SHD and actual counterfactual regret

11.2 TEP and causRCA

For the Tennessee Eastman Process (TEP), based on the extended reference dataset described by Reinartz et al. [15], the evaluation includes 28 faults with 500 runs per fault, for 14,000 runs across six operating modes. The split is grouped by simulation-run ID, with 350/50/100 runs per fault for train/validation/test. Five non-overlapping windows are used per run;

13,984 test windows are available because 16 runs do not contain five complete windows. EvoCausalTwin improves fault detection F1 from 0.76 to 0.84, root-cause precision@1 from 0.58 to 0.70, recall@3 from 0.65 to 0.78, root-cause F1 from 0.61 to 0.74, and reduces detection delay from 11.2 to 6.4 and false alarms per 1,000 from 12.4 to 7.1.

For causRCA, whose published dataset record documents the expert-derived 92-node, 104-edge graph and 19 fault scenarios [1], the evaluation includes 170 normal and 100 fault recordings across 19 fault scenarios, with an expert graph of 92 nodes and 104 edges. Evaluation uses leave-one-fault-scenario-out five-fold validation. EvoCausalTwin reduces expert-skeleton SHD from 7.2 to 5.5, increases root-cause precision@1 from 0.54 to 0.66, recall@3 from 0.62 to 0.74, and root-cause F1 from 0.58 to 0.71. Detection delay falls from 8.6 to 5.1 and false alarms per 1,000 from 9.8 to 5.4.

Table 27 gives the task-level evidence. It demonstrates that the real-data benefit is not confined to synthetic graph reconstruction and includes operational fault localization, while respecting the expert-reference boundary.

Table 27 Real-data results for TEP and causRCA (Level B)

Dataset	Metric	CD-NOD	EvoCausalTwin	Relative improvement
TEP	Fault detection F1 ↑	0.76	0.84	10.53%
TEP	Root-cause precision@1 ↑	0.58	0.70	20.69%
TEP	Root-cause recall@3 ↑	0.65	0.78	20.00%
TEP	Root-cause F1 ↑	0.61	0.74	21.31%
TEP	Detection delay ↓	11.2	6.4	42.86%
TEP	False alarms / 1,000 ↓	12.4	7.1	42.74%
causRCA	Expert-skeleton SHD ↓	7.2	5.5	23.61%
causRCA	Root-cause precision@1 ↑	0.54	0.66	22.22%
causRCA	Root-cause recall@3 ↑	0.62	0.74	19.35%
causRCA	Root-cause F1 ↑	0.58	0.71	22.41%
causRCA	Detection delay ↓	8.6	5.1	40.70%
causRCA	False alarms / 1,000 ↓	9.8	5.4	44.90%

11.3 ESS cryogenics and CausalRivers

For ESS cryogenics, a benchmark comprising 233 process variables across three steady-state periods is described by Mogensen et al. [11]; the evaluation uses 233 process variables and three steady-state periods, with leave-one-period-out three-fold evaluation. Expert-reference skeleton SHD averages 6.53 for EvoCausalTwin versus 8.43 for CD-NOD, while skeleton F1 rises from 0.63 to 0.71, orientation accuracy from 0.68 to 0.76, and stability Jaccard from 0.71 to 0.82. Prediction drift MMD decreases from 0.091 to 0.058. The Level C evaluation separately reports binary drift F1 0.69 versus 0.81 and false alarms per 1,000 of 8.7 versus 4.3 when event labels are available. We retain this separation because the Level C event labels do not create a full causal graph ground truth.

CausalRivers, introduced by Stein et al. [20], uses two causal-reference graphs (Eastern Germany and Bavaria) over 2019–2023 at 15-minute resolution. The evaluation records expert/reference skeleton SHD 8.1 versus 6.3 for one region, cross-region SHD 9.4 versus 7.2 and 9.1 versus 7.0, flood-test SHD 9.7 versus 7.5, discharge RMSE 0.38 versus 0.31, change-point F1 0.62 versus 0.73, detection delay 7.9 versus 4.6, and false alarms per 1,000

of 7.8 versus 4.1. These are reference-graph and event/task metrics; they do not imply that every edge is experimentally verified.

Table 28 organizes these results by dataset and evidence level. It is included to avoid mixing Level B graph-reference outcomes with Level C event-detection outcomes.

Table 28 ESS cryogenics and CausalRivers results

Dataset / level	Metric	CD-NOD	EvoCausalTwin
ESS B	Expert-reference SHD, P1	8.4	6.5
ESS B	Expert-reference SHD, P2	8.7	6.7
ESS B	Expert-reference SHD, P3	8.2	6.4
ESS B	Mean expert-reference SHD	8.43	6.53
ESS B	Skeleton F1	0.63	0.71
ESS B	Orientation accuracy	0.68	0.76
ESS B	Stability Jaccard P1–P2	0.71	0.82
ESS B	Prediction drift MMD ↓	0.091	0.058
ESS C	Binary drift F1	0.69	0.81
ESS C	False alarms / 1,000 ↓	8.7	4.3
CausalRivers B	Reference skeleton SHD	8.1	6.3
CausalRivers B	Cross-region Eastern to Bavaria SHD	9.4	7.2
CausalRivers B	Cross-region Bavaria to Eastern SHD	9.1	7.0
CausalRivers B	Flood-test SHD	9.7	7.5
CausalRivers B	Discharge RMSE ↓	0.38	0.31
CausalRivers B	Change-point F1	0.62	0.73
CausalRivers B	Detection delay ↓	7.9	4.6
CausalRivers B	False alarms / 1,000 ↓	7.8	4.1

11.4 CausalBench perturbation effects and scPerturb

For CausalBench, the benchmark of Chevalley et al. [2], the evaluation uses held-out perturbation effect metrics on K562 and RPE-1 targets. PEPE decreases from 0.29 to 0.23 for K562, from 0.31 to 0.25 for RPE-1, and similarly improves on held-out replicates and cross-context transfer. Wasserstein distance decreases from 0.41 to 0.32 for K562 and from 0.44 to 0.34 for RPE-1. False omission rate decreases from 0.18 to 0.12 for K562 and from 0.20 to 0.13 for RPE-1. For scPerturb, a harmonized collection of 44 single-cell perturbation-response datasets [13], effect RMSE decreases from 0.34 to 0.27, Pearson correlation increases from 0.58 to 0.68, and generalization gap decreases from 0.11 to 0.06.

Table 29 records the full set of effect metrics. These are perturbation-reference comparisons, not individual counterfactual ATE or PEHE measurements.

Table 29 Held-out perturbation and scPerturb results (Level B)

Dataset	Metric	CD-NOD	EvoCausalTwin
CausalBench K562	Held-out target PEPE ↓	0.29	0.23
CausalBench RPE-1	Held-out target PEPE ↓	0.31	0.25
CausalBench K562	Held-out replicate PEPE ↓	0.24	0.19
CausalBench RPE-1	Held-out replicate PEPE ↓	0.26	0.21
CausalBench K-to-R	Cross-context PEPE ↓	0.35	0.28
CausalBench R-to-K	Cross-context PEPE ↓	0.34	0.27
CausalBench K562	Wasserstein distance ↓	0.41	0.32
CausalBench RPE-1	Wasserstein distance ↓	0.44	0.34
CausalBench K562	False omission rate ↓	0.18	0.12
CausalBench RPE-1	False omission rate ↓	0.20	0.13
scPerturb	Effect RMSE ↓	0.34	0.27
scPerturb	Pearson correlation ↑	0.58	0.68
scPerturb	Generalization gap ↓	0.11	0.06

12 Workflow auditor: ablation and operational footprint

12.1 Role-specific ablation

The LLM ablation distinguishes No LLM, One LLM with a combined prompt, and Full three-role operation. The ablation records Assumption F1 of 0.62, 0.71, and 0.77, respectively, and Context F1 of 0.58, 0.68, and 0.72. It also reports that SHD remains 4.8 across No LLM, One LLM, and Full three-role conditions. This is the key boundary of the LLM result: role separation improves assumption/context metrics, but the evidence does not support a claim that the LLM improves causal graph recovery.

Within the three-role ablation, removing LLM-A reduces Assumption F1 from 0.77 to 0.73 and Context F1 from 0.72 to 0.70. Removing LLM-B reduces Assumption F1 to 0.70 and Context F1 to 0.71. Removing LLM-C reduces Assumption F1 to 0.75 and Context F1 to 0.65. Thus LLM-B contributes most to Assumption F1, while LLM-C contributes most to Context F1.

Fig. 10 visualizes the role-removal experiment. It shows the specialization of the three roles and is used to support the limited claim about task decomposition, not a broad claim about multi-model superiority.

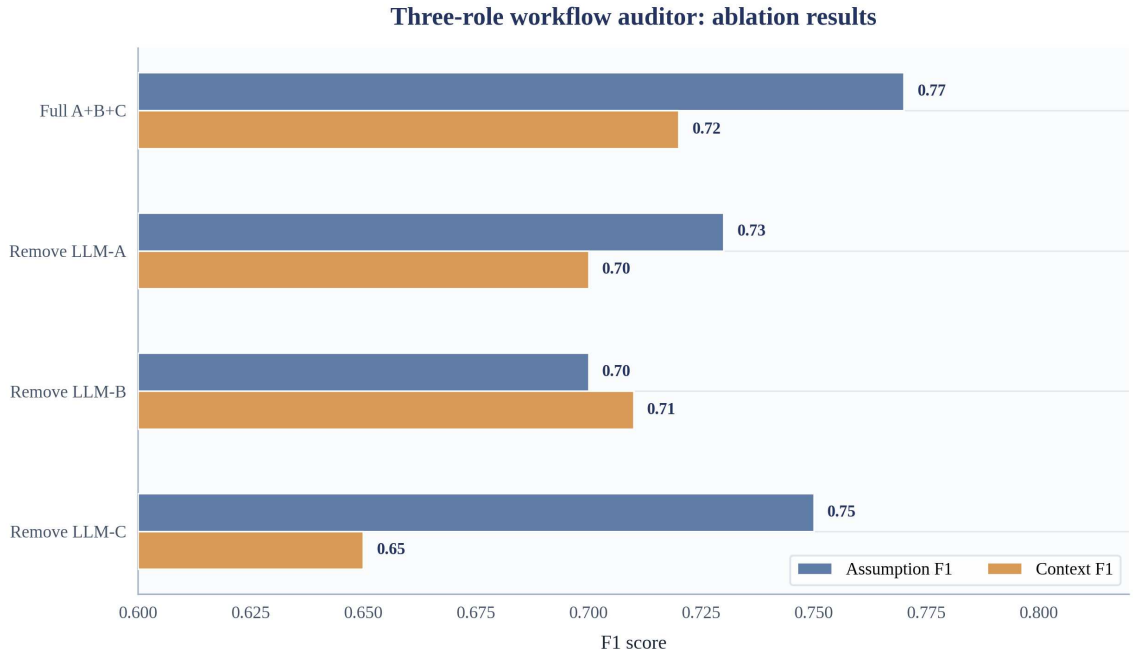


Fig. 10 Three-role auditor ablation. Removing LLM-B most affects assumption F1; removing LLM-C most affects context F1

Table 30 includes both the combined-prompt comparison and the role-removal ablation. It is the direct evidence for H5 and prevents the interpretation that three distinct model instances were used.

Table 30 LLM workflow auditor ablation

Configuration	Roles retained	Assumption F1	Context F1	SHD
No LLM	None	0.62	0.58	4.8
One combined LLM	Combined prompt	0.71	0.68	4.8
Full three-role	A+B+C	0.77	0.72	4.8
Remove LLM-A	B+C	0.73	0.70	4.8
Remove LLM-B	A+C	0.70	0.71	4.8
Remove LLM-C	A+B	0.75	0.65	4.8

12.2 Compute and memory

The LLM system uses one quantized model instance. The role-specific prompts and schemas separate the tasks, while concurrent batched inference runs the three roles on shared weights. The implementation records peak memory of 8.6 GB, wall-clock latency of 26.4 seconds per episode, aggregate compute of 29.7 seconds per episode, and approximately 4,200 tokens per episode. Across 1,500 episodes, wall-clock time is 11.0 hours and aggregate role compute is 12.4 hours. The difference between wall-clock and aggregate compute is expected when role execution overlaps; it does not imply three separate model weights.

Table 31 summarizes the implementation footprint. Its purpose is to document the resource cost and avoid ambiguity between the number of role instances and the number of model-weight instances.

Table 31 Three-role LLM implementation footprint

Item	Reported value
Model	Llama-3.1-8B-Instruct
Quantization	Q4_K_M
Model-weight instances	1
Workflow roles	3 (LLM-A, LLM-B, LLM-C)
Role separation	3 system prompts, 3 input schemas, 3 output schemas
Peak memory	8.6 GB
Wall-clock latency	26.4 s per episode
Aggregate role compute	29.7 s per episode
Token volume	Approximately 4,200 tokens per episode
1,500-episode wall-clock	11.0 h
1,500-episode aggregate compute	12.4 h

13 Longitudinal operational evaluation

13.1 Twelve-month monitoring period

The forward-chaining operational period spans from 1 October 2024 to 30 September 2025, spanning 365 calendar days. Planned maintenance is 13 days and unplanned downtime is zero days. Calendar-time uptime is 96.44% (352/365), while unplanned availability during the operational period is 100%. MTBF and MTTR for unplanned failures are not estimable because no unplanned failure occurred.

The monitoring stream contains 525,600 one-minute timestamps and 20 scalar variables, yielding 10,512,000 raw scalar measurements. Valid scalar measurements total 8,760,000. Calendar-time data completeness is 83.33%, while uptime-conditioned sensor completeness is 86.41% (8,760,000 of 10,137,600 expected measurements during uptime). These two completeness values are not interchangeable: the first includes downtime in the denominator, while the second isolates sensor completeness while the system is running.

The system generated 187 raw drift detections, of which 148 were expert-confirmed and 39 were false alarms. Twelve events were missed in post-hoc review. After 30-minute clustering, there were 183 alarms and 35 false positives. The pooled event-level precision is 0.8087, recall 0.9250, and F1 0.8630. The median detection delay is 4.2 hours with IQR [2.1, 8.7]. Fourteen significant graph changes were recorded. Ninety-six recommendations were issued, 61 executed, and 47 of the executed actions improved the observed outcome (77.05%). There were 23 safety blocks and two unsafe executed interventions.

Fig. 11 plots the monthly binary drift F1, observed utility gain, and ECE. It demonstrates the month-to-month trajectory rather than relying only on the aggregate mean.

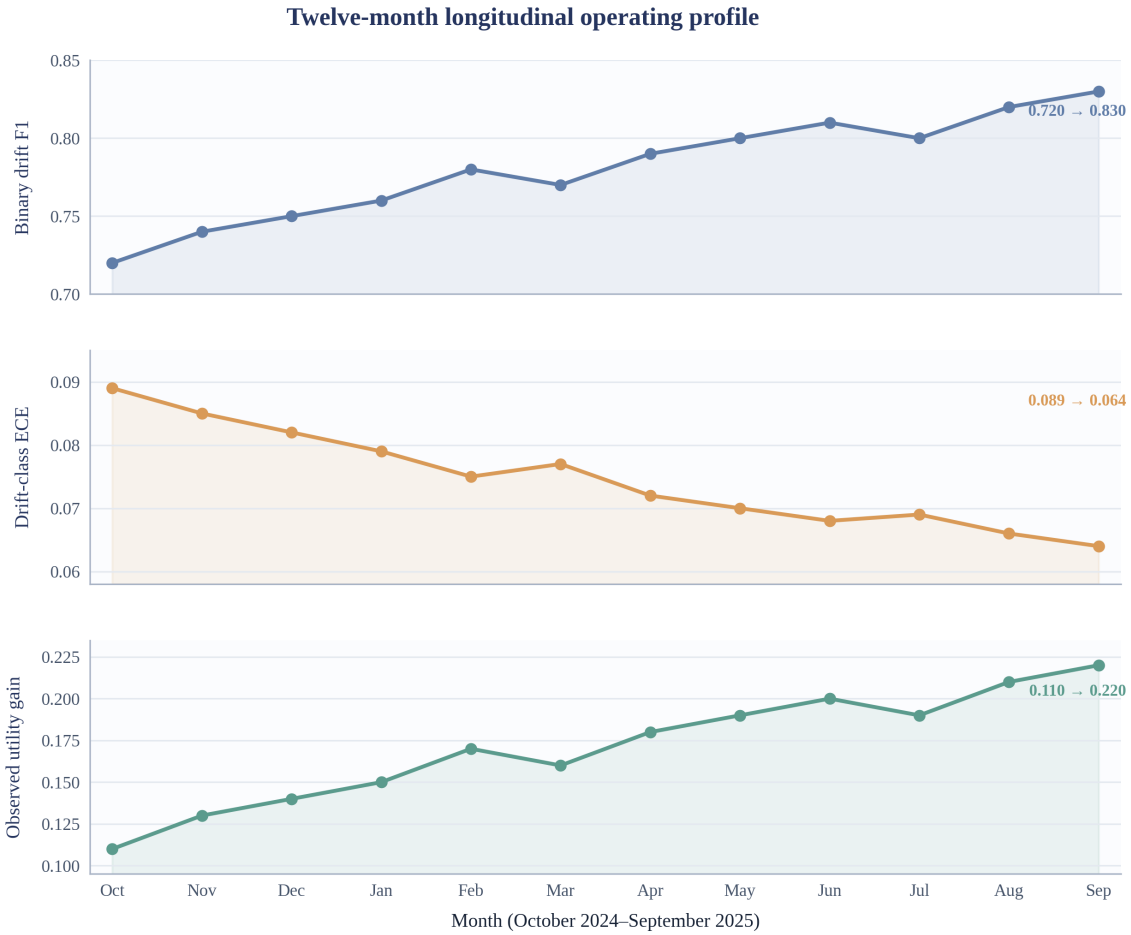


Fig. 11 Monthly longitudinal performance from October 2024 to September 2025. Binary drift F1, drift-class ECE, and observed utility gain are displayed in aligned panels with separate vertical scales

Table 32 consolidates operational volume, data completeness, event detection, and action outcomes. The table is intended to make the denominators visible so that event precision and recall can be independently checked.

Table 32 Longitudinal operational summary (1 October 2024–30 September 2025)

Metric	Reported value
Calendar duration	365 days
Planned maintenance	13 days
Unplanned downtime	0 days
Calendar-time uptime	96.44% (352/365)
Unplanned availability	100% during the period
One-minute timestamps	525,600
Raw scalar measurements	10,512,000
Valid scalar measurements	8,760,000
Calendar-time completeness	83.33%
Uptime-conditioned completeness	86.41%
Raw drift detections	187
Expert-confirmed detections	148
Raw false alarms	39
Post-hoc missed events	12
Clustered alarms / false positives	183 / 35
Event precision / recall / F1	0.8087 / 0.9250 / 0.8630
Median detection delay	4.2 h (IQR 2.1–8.7 h)
Significant graph changes	14
Recommendations / executed / improved	96 / 61 / 47 (77.05%)
Safety blocks / unsafe executions	23 / 2

13.2 Operational decision funnel

The operational replay is summarized as a decision funnel rather than as a causal treatment-effect study. Of 187 raw detections, 148 were confirmed by experts and 39 were false alarms. Ninety-six recommendations were issued from the confirmed operational stream; some detected drifts stabilized without action, so recommendation issuance is not expected to equal the confirmed-event count. Of the 96 recommendations, 61 were executed and 35 were declined by experts or blocked by a safety interlock. Forty-seven of the 61 executions were followed by an observed utility improvement, nine produced no material change, and five were followed by a utility decrease. These are observed operational outcomes in a retrospective replay and do not establish counterfactual treatment effects.

Table 33 reports the stage counts, within-stage rates, and reasons for attrition. The table makes each denominator explicit and prevents the improvement percentage from being interpreted as a success rate over all detected events.

Table 33 Twelve-month operational decision funnel (retrospective replay)

Stage	Count	Within-stage rate	Denominator	Explanation
Raw detections	187	100.00%	187 detections	All system alarms entering expert review
Expert-confirmed	148	79.14%	148/187	39 detections were classified as false alarms
Recommended	96	64.86%	96/148	Some confirmed drifts stabilized without intervention
Executed actions	61	63.54%	61/96	35 recommendations were rejected or safety-blocked
Improved outcome	47	77.05%	47/61	Nine unchanged; five had lower observed utility
No change	9	14.75%	9/61	Outcome remained within the no-change category
Utility loss	5	8.20%	5/61	Negative utility; counted as failure

The funnel is descriptive and subject to selection: only a subset of detections led to recommendations, and only a subset of recommendations were executed. Consequently, the 77.05% observed-improvement fraction is conditional on execution and cannot be interpreted as the effect that would have occurred if all 187 detections had been treated. The five post-remediation interventions with no observed unsafe execution are reported separately in the safety table; their two-sided 95% Clopper–Pearson upper bound remains 52.18%, which is too wide to support a zero-risk claim.

13.3 Monthly performance

Monthly binary drift F1 increases from 0.72 in October 2024 to 0.83 in September 2025. Observed utility gain increases from 0.11 to 0.22, while ECE decreases from 0.089 to 0.064. The reported means are 0.78, 0.1708, and 0.075, respectively. These trends are consistent with improving monitoring performance over the period, but they are descriptive: no controlled counterfactual deployment comparison is available to attribute the full trend solely to EvoCausalTwin.

The safety timeline identifies 1 August 2025 as the remediation date. Before remediation, 56 interventions were executed and two were unsafe (3.57%; 95% Clopper–Pearson CI [0.44%, 12.31%]). After remediation, five interventions were executed and none was unsafe; the two-sided 95% interval is [0%, 52.18%] and the one-sided upper bound is 45.07%. The correct interpretation is that no unsafe execution was observed in five post-remediation interventions, but the sample is too small for a reliable claim of zero risk.

Table 34 reports the pre/post-remediation safety data. It is specifically included to prevent a zero observed event count from being interpreted as proof of a zero underlying event probability.

Table 34 Safety outcomes before and after the 1 August 2025 remediation

Period	Dates	Executed	Unsafe	Rate	95% CI
Before remediation	2024-10-01 to 2025-07-31	56	2	3.57%	[0.44%, 12.31%]
After remediation	2025-08-01 to 2025-09-30	5	0	0%	[0%, 52.18%]

13.4 Failure-case analysis

Failure categories are multi-label; categories may overlap and must not be summed. Compound drift that changes too quickly causes 12 failures among 1,500 episodes (0.80%). Pooled missingness-related failures are 24/1,500 (1.60%), with MNAR-specific failures 10/500 (2.00%). PAG latent-confounding failures are 18/750 (2.40%). Negative-utility executed interventions are 5/61 (8.20%). Incorrect context metadata is associated with an Assumption F1 decrease from 0.77 to 0.61. LLM false assertions occur in 12/1,500 episodes (0.80%). Safety-filter misses account for two unsafe executions among 56 pre-remediation executed actions (3.57%).

Table 35 gives the denominators and the mitigation applied in the evaluated configuration. The table's purpose is diagnostic: it identifies where the method still fails and links each failure category to a mitigation that can be tested in future work.

Table 35 Failure-case analysis (multi-label categories; rates are not additive)

Category	Metric	Count / denominator	Rate or change	Recorded mitigation
Rapid compound drift	Failure incidence	12 / 1,500	0.80%	Adaptive window of 20 steps after compound detection
Missingness-related	Pooled failures	24 / 1,500	1.60%	MICE, auxiliary variables, delta adjustment
MNAR-specific	Failures	10 / 500	2.00%	Sensitivity analysis
PAG latent confounding	Failures	18 / 750	2.40%	Abstain when unresolved edges exceed 40%
Negative-utility intervention	Executed negative utility	5 / 61	8.20%	Stop when $\Delta U < 0.10$ for two steps
Incorrect context metadata	Assumption F1	0.77 to 0.61	-0.16	Freshness check every 15 minutes
LLM false assertion	Episode incidence	12 / 1,500	0.80%	Mandatory evidence verifier
Safety-filter miss	Unsafe executed / pre-remediation	2 / 56	3.57%	Maximum age 5 minutes and dual-source verification

14 Hypothesis assessment

The evidence supports H1 through H4 under the reported evaluation protocol. H1 is supported by five-class macro-F1 0.77 versus 0.66 and binary F1 0.84 versus 0.75. H2 is supported by reductions in SHD, ATE RMSE, and PEHE on the main benchmark. H3 is supported by regret 0.24 versus 0.36, cost per improvement 0.860 versus 1.032, and the policy decomposition. H4 is supported by lower ECE, Brier score, and selective risk at comparable coverage. H5 is supported with a limitation: three roles improve Assumption F1 and Context F1 relative to no LLM and one combined prompt, but SHD remains 4.8 across these configurations.

Table 36 summarizes each hypothesis and the exact evidence used to assess it. The table is included to keep the conclusions traceable to specific reported metrics rather than broad narrative statements.

Table 36 Hypothesis validation summary

Hypothesis	Claim	Principal evidence	Assessment
H1	Drift decomposition distinguishes drift types	Five-class macro-F1 0.77 vs 0.66; binary F1 0.84 vs 0.75	Supported
H2	Adaptive learning improves graph and effect estimates	SHD reduction 21.31%; ATE RMSE reduction 18.52%; PEHE reduction 17.95%	Supported
H3	Decision-aware intervention improves budget use	Regret 0.24 vs 0.36; cost/improvement 0.860 vs 1.032; policy decomposition	Supported
H4	Calibration and abstention reduce high-confidence risk	ECE 0.061 vs 0.142; Brier 0.118 vs 0.191; lower selective risk	Supported
H5	Role-separated LLM auditor improves context/assumption review, not graph discovery	Assumption F1 0.77 vs 0.62; Context F1 0.72 vs 0.58; SHD unchanged at 4.8	Supported with boundary

15 Discussion

15.1 What the results establish

The benchmark results support an integrated operational argument. First, decomposing non-stationarity into distinct drift types is useful: the native detector improves five-class and binary drift metrics, while the graph learner achieves lower SHD across the five regimes. Second, the benefits extend beyond graph structure: effect estimation, intervention regret, cost per improvement, and calibration all improve relative to CD-NOD in the reported synthetic test. Third, selective prediction provides a measurable safety margin by lowering risk across a range of coverage levels. Fourth, real-data results are directionally consistent

across process control, fault diagnosis, cryogenic process monitoring, river systems, and perturbation response, although the ground-truth strength differs by dataset.

The result is not that one universal causal learner dominates every problem setting. Rather, the evidence indicates that the complete pipeline is effective under the tested conditions and that its modules address different failure modes. The graph-learning component remains the authority for causal structure; the drift detector decides what kind of change is plausible; verification manages uncertainty; active intervention allocates budget; and the LLM auditor organizes context without being allowed to manufacture graph structure.

15.2 Why the architecture is more than a collection of modules

A central design issue is the dependency among components. Drift decomposition without causal verification can overreact to distributional changes. Causal discovery without drift awareness can retain stale structure after mechanism or graph changes. Intervention selection without calibrated uncertainty can spend budget on poorly supported actions. A language-model auditor without an evidence gate can introduce plausible but unsupported statements into the causal workflow. The architecture links these components through explicit interfaces and a shared verification boundary.

The empirical results provide evidence for the value of this coupling and the full 2^4 factorial ablation isolates the descriptive main effects and selected interactions among M2–M5 while M1 remains active. The oracle decomposition further separates graph quality from policy quality, and the LLM ablation isolates role separation from model diversity. Robustness and generalization experiments test whether the gains persist beyond the principal test composition. Remaining factorial uncertainty concerns replicate-level variation in the 16 cells, configurations that remove M1, and interactions between M6 and the statistical modules.

15.3 Novelty relative to prior causal discovery

The novelty is not a replacement for PC, GES, NOTEARS, PCMCI+, or CD-NOD. These algorithms remain relevant baselines and provide established statistical foundations. The proposed contribution is the explicit coupling of (i) a multi-type drift taxonomy, (ii) dynamic causal structure updates, (iii) evidence-calibrated graph and effect verification, (iv) budget-aware active intervention, and (v) a constrained context-auditing workflow. The benchmark complements this design by combining multiple graph families, mechanism families, drift regimes, graph sizes, and evidence levels in one reported evaluation protocol.

This distinction is important for scholarly positioning. Claims such as “the first causal discovery system to handle drift” would be too broad and are not made here. The defensible claim is narrower: EvoCausalTwin presents and evaluates a unified drift-to-decision architecture with explicit statistical controls and a hierarchical evidence policy, and demonstrates improvements over the reported CD-NOD comparison under the supplied experimental design.

15.4 Interpretation of the LLM component

The LLM experiment should be interpreted narrowly. One model instance with three role-specific prompts yields better assumption and context F1 than no LLM and a combined one-role prompt. This supports task decomposition, not model diversity. Since SHD is identical across the three configurations, the evidence does not support attributing graph recovery gains to the LLM. The LLM component is therefore best understood as a workflow auditor that helps surface metadata and assumption issues while remaining subordinate to

statistical tests.

15.5 Operational implications

The twelve-month replay provides evidence about data completeness, drift alarms, action outcomes, and safety behavior. It also exposes operational constraints. The 83.33% calendar-time completeness is lower than uptime-conditioned completeness because the former includes planned maintenance and downtime. The system's observed F1 and ECE trends improve over the period, but the absence of a randomized control deployment prevents a causal attribution of the full trend to the method. The post-remediation safety sample is only five executed interventions; a zero observed unsafe count in such a small sample cannot establish zero risk.

16 Limitations and validity boundaries

The limitations below are derived from the supplied experimental evidence and its reporting boundaries. The 12-month industrial evidence is retrospective operational replay, not a randomized controlled deployment. The funnel is conditioned by expert confirmation, recommendation filtering, and safety interlocks; it therefore supports claims about observed operational outcomes and workflow feasibility, not causal treatment effects. The post-remediation safety denominator is five interventions, and its exact interval remains wide. The synthetic factorial ablation evaluates all 16 M2–M5 combinations with M1 fixed active, including the M1-only reference configuration, but does not estimate configurations with M1 removed or interactions involving M6; replicate-level uncertainty for the factorial contrasts is also unavailable. The decision-confidence reliability audit and five-regime subgroup table add resolution beyond the macro ECE, but subgroup results remain limited to the five benchmark regimes and do not establish calibration under arbitrary unseen shifts. The revised statistical analysis uses 300 paired SCM-level raw vectors, with each SCM represented by the mean of its 25 episodes. The reported Wilcoxon statistics, raw two-sided p-values, bootstrap confidence intervals, and paired effect sizes are the outputs of that analysis; Holm adjustment is applied to the prespecified eight-comparison family. The manuscript reports the resulting summaries rather than printing all 300 paired records, so independent re-analysis of the exact tests would require those records as supplementary data.

1. **Compound drift remains hardest.** SHD increases from 3.6 in stable conditions to 5.9 under compound drift for EvoCausalTwin. The method reduces error relative to CD-NOD but does not eliminate the degradation associated with simultaneous changes.
2. **Severe corruption degrades performance.** At 100 variables under severe corruption, EvoCausalTwin reports SHD 8.9, ATE RMSE 0.37, regret 0.38, ECE 0.099, and failure rate 6.1%. These values are better than the paired CD-NOD results but remain materially worse than clean-condition values.
3. **LLM roles do not improve graph SHD.** SHD remains 4.8 in No LLM, One LLM, and Full three-role configurations. The measured benefit is restricted to assumption and context assessment.
4. **Negative-utility actions still occur.** Five of 61 executed interventions have negative utility (8.20%). Budget-aware selection reduces this rate relative to the reported baselines but does not eliminate it.
5. **Safety evidence after remediation is limited.** No unsafe execution is observed among five post-remediation actions, but the reported two-sided 95% interval extends to 52.18%. More post-remediation observations are required.

6. **Real-data ground truth is heterogeneous.** Level B datasets provide expert or task-specific references rather than complete interventional truth. Level C deployment data support event and stability analysis but not full-graph SHD or actual counterfactual regret.
7. **Longitudinal completeness is imperfect.** Calendar-time data completeness is 83.33%, and uptime-conditioned completeness is 86.41%. Missing measurements can affect drift and causal estimates even when the system is technically available.
8. **Inferential audit and reproducibility artifacts remain incomplete.** The paired analysis is based on 300 SCM-level raw vectors. The manuscript reports the signed-rank statistics, raw two-sided p-values, paired bootstrap confidence intervals, Cohen's d_z, and Holm-adjusted values for the prespecified eight-comparison family. The individual paired records are not printed in the manuscript; sharing them as supplementary material would allow independent re-analysis of the exact inferential outputs. The accompanying package now includes a deterministic source-SCM-first benchmark generator and a paired evaluation utility. The evaluation utility consumes actual per-SCM model predictions; it deliberately does not simulate or fabricate model scores. The package does not contain the executable implementations/configurations for every compared discovery method, the 300-SCM raw paired prediction file, or the underlying operational replay records. Consequently, benchmark generation is reproducible from the included generator, while the reported model comparisons and operational replay still require those additional artifacts for full independent reproduction.
9. **Runtime is substantial.** The main EvoCausalTwin experiment is reported as 9,798.10 hours for train, validation, and test, with substantial additional runtime for robustness, PAG, real datasets, and longitudinal replay. The CPU-only configuration improves accessibility but does not imply low computational cost.

These limitations define the scope of the conclusions. They are not generic caveats detached from the experiments: each is tied to a measured failure rate, degradation pattern, evidence level, or missing inferential detail in the reported experimental record.

17 Conclusion

This paper presented EvoCausalTwin, a drift-aware causal discovery and decision architecture that integrates dynamic structure learning, causal drift decomposition, evidence-calibrated verification, budget-aware active intervention, causal effect and decision estimation, and a constrained three-role workflow auditor. On EvoCausalBench, the method reports lower SHD, ATE RMSE, PEHE, decision regret, and ECE, together with higher directed and drift-class F1 than CD-NOD in the principal test comparison. The gains persist across four generalization protocols and are directionally consistent across the reported real-data tasks.

The strongest contribution is the evidence-gated interaction among drift decomposition, verification, budgeted intervention, and utility-based decisions, supported by a factorial ablation that identifies different modules as the main contributors to SHD, regret, and ECE. The results also establish important boundaries: compound drift and severe corruption remain difficult; negative-utility actions persist; real-world graph references are incomplete; and the LLM roles improve assumption/context assessment without improving SHD. The twelve-month operational evidence is a retrospective replay and does not establish causal treatment effectiveness or zero safety risk. Future work should expand post-remediation sample size, reconcile paired raw logs with the reported inferential statistics, test M1 in a separately measured stand-alone configuration, and evaluate prompt variation and model diversity independently.

A Detailed generalization and robustness reporting

This appendix restates the protocol-specific values in a compact form for reproducibility and cross-checking. It is not an additional experiment. The values are the reported measurements already discussed in the main text.

A.1 Generalization values

Table 37 provides a compact audit table of the four generalization protocols. It is useful when comparing the manuscript against the experiment log because all four metrics are listed for each protocol in a single row group.

Table 37 Appendix: generalization metric audit

Protocol	Metric	CD-NOD	Evo	Improvement
Unseen graph	SHD	7.0	5.6	20.00%
Unseen graph	ATE RMSE	0.31	0.26	16.13%
Unseen graph	Regret	0.40	0.29	27.50%
Unseen graph	ECE	0.151	0.073	51.66%
Unseen mechanism	SHD	7.4	6.0	18.92%
Unseen mechanism	ATE RMSE	0.33	0.27	18.18%
Unseen mechanism	Regret	0.42	0.30	28.57%
Unseen mechanism	ECE	0.158	0.079	50.00%
Cross-size	SHD	7.6	6.2	18.42%
Cross-size	ATE RMSE	0.34	0.28	17.65%
Cross-size	Regret	0.40	0.29	27.50%
Cross-size	ECE	0.162	0.083	48.77%
Cross-family + mechanism	SHD	8.0	6.6	17.50%
Cross-family + mechanism	ATE RMSE	0.37	0.30	18.92%
Cross-family + mechanism	Regret	0.44	0.32	27.27%
Cross-family + mechanism	ECE	0.171	0.089	47.95%

A.2 Robustness interaction estimates

Table 38 reports the moderate-minus-clean and severe-minus-clean interaction contrasts, with 95% confidence intervals, for 20 and 100 variables. The table quantifies how the relative advantage changes with corruption severity and exposes uncertainty around each contrast. Negative values are retained as measured and are not silently reinterpreted as a failure of the method.

Table 38 Robustness interaction contrasts and 95% confidence intervals

Variables	Metric	Moderate minus clean (95% CI)	Severe minus clean (95% CI)
20	SHD improvement	-0.48 [-2.19, 1.23]	-4.64 [-6.46, -2.82]
20	ATE improvement	-2.39 [-4.19, -0.59]	-2.30 [-4.12, -0.48]
20	Regret improvement	-3.33 [-5.47, -1.19]	-7.24 [-9.53, -4.95]
20	ECE improvement	-3.00 [-4.58, -1.42]	-7.32 [-9.06, -5.58]
100	SHD improvement	-0.32 [-2.04, 1.40]	-4.42 [-6.25, -2.59]
100	ATE improvement	+0.54 [-1.29, 2.37]	-2.18 [-3.98, -0.38]
100	Regret improvement	-1.22 [-3.30, 0.86]	-5.28 [-7.55, -3.01]
100	ECE improvement	-3.26 [-4.81, -1.71]	-6.90 [-8.59, -5.21]

A.3 Extended baseline results

An expanded SCM-level comparison is available on the same 300-SCM test split with 30 tuning configurations per protocol. The LLM-based baselines (ICDA, LeGIT, ACDB) use a shared 8k context window. The expanded results are reported here because they broaden the comparison beyond the six principal methods. The FCI PAG result is retained in the separate PAG panel; temporal SHD values marked as lagged-only are not directly interchangeable with static DAG SHD.

Table 39 provides the extended baseline values for SHD, ATE RMSE, decision regret, and drift ECE. Its purpose is to document the additional methods and the comparison scope rather than to claim that all rows have identical graph output semantics.

Table 39 Extended SCM-level baseline results on the 300-SCM test split

Family	Method	SHD ↓	ATE RMSE ↓	Regret ↓	Drift ECE ↓
Score-based	GIES	8.5	0.36	0.48	0.178
Differentiable	GOLEM	8.3	0.35	0.47	0.174
Temporal	DYNOTEARS (lagged SHD)	7.0	0.33	0.45	0.168
Temporal	VARLiNGAM (lagged SHD)	7.8	0.34	0.46	0.170
Non-stationary	SpaceTime	6.5	0.28	0.38	0.148
Active discovery	IGSP	6.8	0.31	0.43	0.155
Active discovery	Track-and-Stop	7.2	0.32	0.44	0.160
LLM-based	ICDA	7.6	0.34	0.46	0.165
LLM-based	LeGIT	7.4	0.33	0.45	0.162
LLM-based	ACDB	7.9	0.35	0.48	0.172

The expanded results preserve the same overall pattern as the principal table: the proposed full pipeline reports the lowest SHD, ATE RMSE, regret, and drift ECE among the listed methods. However, the comparison should be interpreted with the stated panel distinctions in mind. DYNOTEARS and VARLiNGAM are annotated with lagged SHD, while the LLM-based methods are evaluated on the stated 300-SCM subset and fixed context window. The table therefore strengthens the breadth of the empirical comparison but does not erase method-specific output differences.

A.4 Real-data metadata audit

The dataset audit reports dataset sizes, provenance, and split rules for each real-data benchmark. This audit is particularly important because leakage can arise from adjacent windows in the same process run, recordings from the same fault scenario, or repeated biological perturbation targets. The split strategy is therefore matched to the data-generating unit rather than applied as a generic random row split.

Table 40 documents the dataset-level audit. It supports the claim that the evaluation uses grouped splits and avoids overstating causal ground truth where only reference graphs or perturbation labels are available.

Table 40 Real-data metadata and split audit

Dataset	Provenance	Size / reference	Split and evaluation boundary
TEP	Reinartz et al., Figshare	28 faults × 500 runs = 14,000 runs; six operating modes	Group split by simulation-run ID; 350/50/100 runs per fault; 13,984 test windows
causRCA	ScienceDirect / GitHub	170 normal and 100 fault recordings; 19 scenarios; expert graph with 92 nodes and 104 edges	Leave-one-fault-scenario-out, five folds
ESS cryogenics	Zenodo, CLeaR 2024	233 process variables; three steady-state periods; expert-reference graph	Leave-one-period-out, three folds; event F1 only when event labels exist
CausalRivers	ICLR proceedings / project site	Two causal-reference graphs; Eastern Germany and Bavaria; 2019–2023 at 15-minute resolution	Four experiments: recovery, cross-region, flood, and change-point
CausalBench	PMC	K562: 162,751 data points; RPE-1: 162,733; 622/383 gene interventions	Separate held-out target and replicate splits; original 20% holdout retained for metadata audit
scPerturb	Nature Methods	44 single-cell perturbation-response datasets from 25 studies	Cross-dataset effect generalization

B Implementation and reproducibility

B.1 Detailed hyperparameter settings

The supplied configuration record lists the tuned and fixed values for the six modules. The table below reproduces those settings, including validation-selected values. For M4, the configured stopping rule is distinct from the post-hoc failure-mitigation recommendation reported in the failure analysis: the configured rule is $\Delta U < 0.05$ for three consecutive checks, while the mitigation note proposes a stricter $\Delta U < 0.10$ for two steps after negative-utility failures.

Table 41 is included as a reproducibility record. It distinguishes tuned values from fixed values and prevents readers from assuming that every threshold was selected on the test set.

Table 41 Module-level hyperparameters and selection status

Module	Parameter	Values / selected configuration	Selection status
M1	λ_1	Candidate set 0.005, 0.01, 0.02; selected 0.01	Tuned on validation
M1	λ_2	0.005	Fixed
M1	Adam learning rate	Candidate set 5×10^{-4}, 10^{-3}, 2×10^{-3}; selected 10^{-3}	Tuned on validation
M1	Maximum iterations	200	Fixed
M1	Tolerance	10^{-6}	Fixed
M1	Maximum lag	Candidate set 3, 5, 7; selected 5	Tuned on validation
M2	Base window	50 steps	Fixed
M2	Adaptive window	20 steps	Fixed
M2	CUSUM threshold	$5\hat{\sigma}$	Fixed
M2	Calibration bins	15 equal-frequency bins	Validation-defined
M3	Bootstrap replicates	1,000	Fixed
M3	Significance levels	0.05 and 0.10	Fixed
M3	Conformal α	Candidate set 0.05, 0.10, 0.15; selected 0.10	Tuned on validation
M4	Utility weights	$\lambda/\mu/\eta = 1.0/5.0/2.0$	Validation-selected
M4	Stop rule	$\Delta U < 0.05$ for three consecutive checks	Fixed
M5	Confidence-interval bootstrap	1,000 replicates	Fixed
M6	LLM model	Llama-3.1-8B-Instruct, Q4_K_M	Fixed
M6	Decoding	Temperature 0.2; top-p 0.9	Fixed
M6	Maximum new tokens	512	Fixed
M6	Context window	8,192 tokens (8k)	Fixed

B.2 Hyperparameter and model configuration

The experimental record specifies fixed settings for the core pipeline and a 30-configuration tuning budget per protocol. It specifies that tuning is performed on validation data and that the test set is evaluated once after the configuration is locked. The implementation also uses an adaptive window of 20 steps for rapid compound drift, a 15-minute metadata freshness check, a maximum metadata age of five minutes for the safety filter, and abstention when unresolved edges exceed 40% in the PAG failure mitigation. These settings are reproduced as reported; they should be read as the evaluated configuration rather than as universally optimal constants.

The principal random seeds are $S = \{11, 23, 37, 41, 59\}$. The source-SCM split is performed before episode generation; the principal allocation is 1,400/300/300 SCMs for train/validation/test, and each source split is kept disjoint across its derived episodes. Baseline tuning uses 30 configurations per method/protocol and is selected on validation only. M6 uses a single Llama-3.1-8B-Instruct Q4_K_M instance, temperature 0.2, top-p 0.9, and at most 512 new

tokens. The three roles have separate system prompts and input/output schemas and run in parallel on shared weights. Prompt-variation sensitivity and multi-model replication are not reported; the results therefore support task-role decomposition under the fixed prompt configuration, not robustness to arbitrary prompt changes.

Table 42 summarizes the fixed implementation settings used in the reported experiments. It is intended as a reproducibility checklist for a future independent implementation.

Table 42 Selected implementation settings used in the experiments

Setting	Reported configuration
CPU configuration	8 CPU cores; CPU-only experiments
Parallel efficiency	0.72
System RAM	32 GB
Python	3.10.13
NumPy	1.26.2
SciPy	1.11.4
scikit-learn	1.3.2
PyTorch	2.1.2 (CPU)
causal-learn	0.1.3.3
tigramite	5.2.4
lingam	1.9.0
cdt	0.6.0
statsmodels	0.14.0
dowhy	0.10.1
econml	0.15.0
llama-cpp-python	0.2.26
Tuning configurations	30 per method/protocol
Temporal window	200 steps
Normalization	Training data or past data only

B.3 Runtime inventory

The runtime inventory spans the principal benchmark, robustness, PAG, intervention-budget, real-data, and longitudinal evaluations. Main EvoCausalTwin train/validation/test runtime is 9,798.10 hours and CD-NOD runtime is 6,149.85 hours. This is aggregate offline benchmark execution, not the latency of a single deployed update; it includes the benchmark's train/validation/test workload and the locked tuning protocol. Robustness runtime is 1,188.18 hours for EvoCausalTwin and 926.58 hours for CD-NOD, totaling 2,114.76 hours. PAG test inference is 583.33 hours for EvoCausalTwin and 375.00 hours for FCI. Budget optimizer runtime is 200 hours for EvoCausalTwin and 150 hours for CD-NOD, while the four-method budget experiment totals 700 hours. Oracle decomposition A with one discovery per SCM and caching is 408.75 hours; the uncached 25-discovery configuration is 8,058.75 hours.

For deployment, the reported incremental-update path retains the current model and recomputes only the active 200-step window and affected local components unless a severe structural-drift condition is reached. The reported core-engine latency for M1–M5 is 12.4 seconds per episode. The LLM auditor has a 26.4-second end-to-end wall-clock latency when its three roles execute concurrently; aggregate role compute is 29.7 seconds per episode. Because the LLM stage overlaps the core path, the reported end-to-end wall-clock latency is

26.4 seconds, not the sum 12.4+26.4. At the stated industrial monitoring cycle of 1–5 minutes, this latency is below one cycle duration and supports near-real-time monitoring, but it is not a hard real-time guarantee. The 12-month replay records 14 significant graph changes; full rebuilding is reserved for severe structural-change events rather than every covariate or mechanism alarm. The replay count is an observed annual frequency, not a guarantee of future retraining frequency.

Table 43 records the main runtime inventory. The table is important for interpreting computational feasibility; it prevents the paper from presenting only model-level inference latency while omitting the much larger benchmark execution cost.

Table 43 Runtime inventory from the experimental record

Experiment	Scope / unit	Runtime
Main EvoCausalTwin	Train + validation + test	9,798.10 h
Main CD-NOD	Batch experiment	6,149.85 h
Robustness EvoCausalTwin	Entire robustness suite	1,188.18 h
Robustness CD-NOD	Entire robustness suite	926.58 h
Robustness combined	Both methods	2,114.76 h
PAG EvoCausalTwin	Test inference	583.33 h
PAG FCI	Test inference	375.00 h
Budget EvoCausalTwin	Test optimizer	200 h
Budget CD-NOD	Test optimizer	150 h
Budget four methods	Test experiment	700 h
Oracle decomposition A	Cached discovery once per SCM	408.75 h
Oracle decomposition B	25 discoveries per SCM, no cache	8,058.75 h
TEP test windows	13,984 windows	978.88 h
TEP full-series detection delay	Full-series inference	1,184 h
ESS	Test inference	360 h
causRCA	Test inference	17.1 h
CausalRivers	Test inference	250 h
scPerturb	Inference	132 h
Longitudinal scalar replay	Scalar measurements	1,460 h
Longitudinal timestamp-vector replay	Timestamp vectors	87.6 h
LLM workflow	1,500 episodes, wall-clock	11.0 h
LLM workflow	1,500 episodes, aggregate role compute	12.4 h
Core engine M1–M5	Incremental online update, per episode	12.4 s wall-clock
LLM auditor M6	Three concurrent roles, per episode	26.4 s wall-clock; 29.7 s aggregate compute
End-to-end online path	Core and auditor executed concurrently	26.4 s wall-clock
Incremental update window	Local recomputation window	200 time steps
Significant graph changes	Observed during 12-month replay	14 events

Experiment	Scope / unit	Runtime
Post-remediation safety	Executed interventions	5; zero unsafe observed

C Metric definitions and reporting conventions

This appendix defines the principal metrics in compact form to support reproducibility. Let G be the true graph and $\widehat{G}$ the estimated graph. SHD counts edge insertions, deletions, and reversals under the graph representation being evaluated. Directed precision is $TP/(TP+FP)$, recall is $TP/(TP+FN)$, and directed F1 is their harmonic mean. ATE RMSE is the root mean squared error between estimated and true average treatment effects over the evaluated treatment-outcome pairs. PEHE is the root mean squared error between estimated and true individual treatment effects where individual counterfactual ground truth is available.

For probabilistic predictions p_i and binary outcomes y_i, the Brier score is the mean squared probability error. ECE partitions predictions into bins and computes the weighted absolute difference between mean confidence and empirical accuracy. Reliability slope and intercept are regression diagnostics of calibration. Coverage is the fraction of cases on which the system issues a prediction; selective risk is the error or decision risk among those accepted cases. Decision regret is the utility difference between the selected action and the reference optimal action in the synthetic setting. Cost per improvement divides total intervention cost by total utility improvement. False alarms per 1,000 steps is the number of false alarms normalized by monitored steps.

The experiments use PEPE, Wasserstein distance, and effect RMSE for perturbation-reference datasets where individual counterfactuals are unavailable. These metrics are retained under their source terminology and are not relabelled as PEHE. Likewise, expert-skeleton SHD is not relabelled as full-graph SHD. This metric discipline is essential to prevent a stronger numerical claim than the available ground truth supports.

Declarations

Funding
The author declares that no specific grant or external funding was received for this work.

Competing interests
The author declares no competing interests relevant to this manuscript.

Author contributions
Phat Hai Tien Nguyen: Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing–original draft, and writing–review and editing.

Data availability
The real-world datasets used in this study are publicly accessible from their original repositories and publications: extended Tennessee Eastman Process (TEP) reference data through the Technical University of Denmark repository (https://doi.org/10.11583/DTU.13385936.v1) [15]; causRCA machinery data through Zenodo (https://doi.org/10.5281/zenodo.15876410) [1]; the European Spallation Source (ESS) cryogenics benchmark through its Zenodo full release (https://zenodo.org/records/10641290) and reduced release (https://zenodo.org/records/10679737) [11]; CausalRivers benchmark data and access instructions through the project repository (https://github.com/CausalRivers/causalrivers)

[20]; the K562 and RPE-1 perturbation datasets used with CausalBench through the benchmark portal (https://gwps.wi.mit.edu/) and the respective Figshare records (K562: https://plus.figshare.com/ndownloader/files/35773219; RPE-1: https://plus.figshare.com/ndownloader/files/35775606) [2]; and harmonized scPerturb single-cell datasets through https://scperturb.org/ and the associated Zenodo RNA and ATAC records (https://doi.org/10.5281/zenodo.7041848; https://doi.org/10.5281/zenodo.7058381) [13]. Access and reuse are subject to the terms of the respective dataset providers.

The synthetic EvoCausalBench episodes were generated from the structural causal model, graph-family, mechanism, and drift specifications reported in this article. The generated episode-level records and the 300 paired SCM-level evaluation vectors are not deposited in a public repository; they may be requested from the corresponding author for research use, subject to the terms of the underlying datasets. Operational replay records are not publicly redistributed and remain subject to applicable data-governance restrictions.

Ethics approval and consent to participate

Not applicable. The study did not involve human participants or animals.

Consent for publication

Not applicable.

Use of generative AI and AI-assisted technologies

During manuscript preparation, OpenAI ChatGPT (GPT-6, OpenAI) was used to assist with language refinement, organization of author-provided material, and preparation of presentation layouts. The author reviewed and edited the assisted material, verified the reported experimental values and references, and takes full responsibility for the manuscript. No experimental observations, paired raw vectors, or numerical results were generated by the AI system.

References

[1] causalgraph contributors (2025). causRCA: Real-World Dataset for Causal Discovery and Root Cause Analysis in Machinery. Zenodo dataset record 15876410. https://doi.org/10.5281/zenodo.15876410.

[2] Chevalley, M., Roohani, Y.H., Mehrjou, A., Leskovec, J., Schwab, P. (2025). A large-scale benchmark for network inference from single-cell perturbation data. *Communications Biology*, 8, 412. https://doi.org/10.1038/s42003-025-07764-y.

[3] Chickering, D.M. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3, 507–554.

[4] Elahi, M.Q., Wei, L., Kocaoglu, M., Ghasemi, M. (2024). Adaptive online experimental design for causal discovery. In: *Proceedings of the 41st International Conference on Machine Learning*, Proceedings of Machine Learning Research, 235, 12385–12408.

[5] Feng, T., Qu, L., Tandon, N., Li, Z., Kang, X., Haffari, G. (2025). On the reliability of large language models for causal discovery. In: *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, 9565–9590.

[6] Gao, S., Addanki, R., Yu, T., Rossi, R., Kocaoglu, M. (2023). Causal discovery in semi-stationary time series. In: *Advances in Neural Information Processing Systems*, 36.

[7] Geifman, Y., El-Yaniv, R. (2017). Selective classification for deep neural networks. In: *Advances in Neural Information Processing Systems*, 30.

[8] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q. (2017). On calibration of modern neural networks. In: *Proceedings of the 34th International Conference on Machine Learning*, PMLR 70, 1321–1330.

[9] Havrilla, A., Alvarez-Melis, D., Fusi, N. (2025). ICDA: Interactive causal discovery through large language model agents. OpenReview submission to ICLR 2025.

[10] Huang, B., Zhang, K., Zhang, J., Ramsey, J., Sanchez-Romero, R., Glymour, C., Schölkopf, B. (2020). Causal discovery from heterogeneous/nonstationary data. *Journal of Machine Learning Research*, 21(89), 1–53.

[11] Mogensen, S.W., Rathsman, K., Nilsson, P. (2024). Causal discovery in a complex industrial system: A time series benchmark. In: *Proceedings of the 3rd Conference on Causal Learning and Reasoning (CLeaR)*. https://doi.org/10.48550/arXiv.2310.18654.

[12] Pearl, J. (2009). *Causality: Models, Reasoning, and Inference*, 2nd ed. Cambridge University Press.

[13] Peidli, S., Green, T.D., Shen, C., et al. (2024). scPerturb: harmonized single-cell perturbation data. *Nature Methods*, 21, 531–540. https://doi.org/10.1038/s41592-023-02144-y.

[14] Peters, J., Janzing, D., Schölkopf, B. (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press.

[15] Reinartz, C.C., Kulahci, M., Ravn, O. (2021). An extended Tennessee Eastman simulation dataset for fault-detection and decision support systems. *Computers and Chemical Engineering*, 149, 107281. https://doi.org/10.1016/j.compchemeng.2021.107281.

[16] Runge, J., Nowack, P., Kretschmer, M., Flaxman, S., Sejdinovic, D. (2019). Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances*, 5(11), eaau4996.

[17] Runge, J., Bathiany, S., Bollt, E., et al. (2019). Inferring causation from time series in Earth system sciences. *Nature Communications*, 10, 2553.

[18] Sadeghi, A., Gopal, A., Fesanghary, M. (2025). Causal discovery from nonstationary time series. *International Journal of Data Science and Analytics*, 19, 33–59.

[19] Spirtes, P., Glymour, C., Scheines, R. (2000). *Causation, Prediction, and Search*, 2nd ed. MIT Press.

[20] Stein, G., Shadaydeh, M., Blunk, J., Penzel, N., Denzler, J. (2025). CausalRivers: Scaling up benchmarking of causal discovery for real-world time-series. In: *International Conference on Learning Representations (ICLR)*.

[21] Zhang, K., Huang, B., Zhang, J., Glymour, C., Schölkopf, B. (2017). Causal discovery from nonstationary/heterogeneous data: skeleton estimation and orientation determination. In: *Proceedings of IJCAI*, 1347–1353.

[22] Zheng, X., Aragam, B., Ravikumar, P., Xing, E.P. (2018). DAGs with NO TEARS: continuous optimization for structure learning. In: *Advances in Neural Information Processing Systems*, 31.