AgentCBO: Causal Belief Routing for Bayesian Optimization under Unknown Graphs

Qiyu Wei[1] **Xin Zhang**[1] **Richard Allmendinger**[2] **Mauricio A. Álvarez**[1]
[1]Department of Computer Science
[2]Alliance Manchester Business School
The University of Manchester, Manchester, UK
{qiyu.wei,xin.zhang-41}@postgrad.manchester.ac.uk
{richard.allmendinger,mauricio.alvarezlopez}@manchester.ac.uk

Abstract

Causal Bayesian optimization (CBO) typically relies on the unrealistic assumption of a perfectly known causal graph. Recent methods relax this constraint by coupling optimization with structure learning, but they often waste sample budgets on structural uncertainties that are irrelevant to the optimizer. In this paper, we formalize *routing-equivalence* to shift the objective from full causal-graph recovery to causal belief routing, where structural differences only matter when they alter optimizer-facing objects such as admissible exploration sets and *do*-priors. We establish conditional fixed-update decision preservation and quantify CEI stability under predictive perturbations on a common exploration-set family. Furthermore, we propose AGENTCBO, a graph-belief routing framework that incorporates LLM causal priors through guarded proposal and commit mechanisms. Specifically, a LANGUAGE-PRIOR AGENT combines LLM beliefs with constraint-based discovery to extract a protected causal backbone alongside residual uncertain edges. A CALIBRATION AGENT and a RESIDUAL REPAIR AGENT then refine only these uncertain structures through score-gated bounded local edits. Finally, a deterministic commit gate evaluates candidate updates using routing-proxy checks before allowing them to influence intervention selection. Across ten benchmarks, AGENTCBO performs comparably to known-graph CBO, demonstrating that guarded LLM priors can improve fixed-budget CBO performance.

1 Introduction

In scientific research, optimizing complex objectives frequently requires carefully planned sequential experiments because evaluating candidate interventions is typically expensive [Jones et al., 1998, Shahriari et al., 2016, Frazier, 2018]. Bayesian optimization (BO) provides a principled framework for this setting by using probabilistic surrogate models and sequential acquisition rules to select informative experiments [Mockus, 1974, Jones et al., 1998, Shahriari et al., 2016, Frazier, 2018]. When causal structure is available, causal Bayesian optimization (CBO) can further improve sample efficiency by exploiting the graph to restrict interventions and initialize graph-informed surrogate models [Pearl, 2009, Aglietti et al., 2020]. Concretely, the causal graph routes the optimizer by determining which variables should be considered for intervention, which exploration sets (ESs) are admissible, and which *do*-calculus priors inform the corresponding surrogate models, ultimately shaping the acquisition policy used at each CBO update.

A long-standing challenge in CBO is that the known-graph assumption is rarely realistic in scientific investigations. Recent unknown-graph CBO methods address this limitation by coupling causal optimization with structure uncertainty. Causal Entropy Optimization models uncertainty over causal graphs within the surrogate and acquisition process to trade off structure learning and causal-effect

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

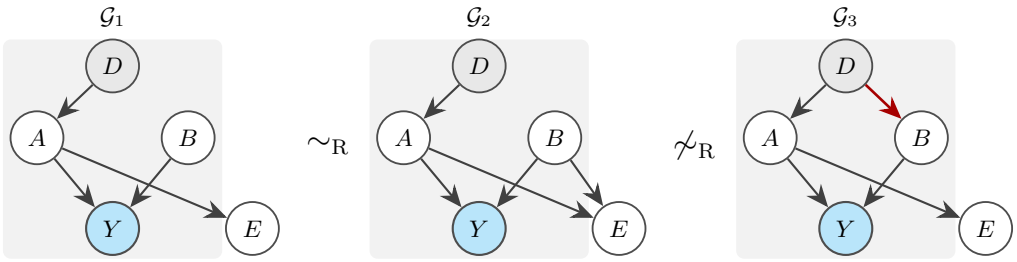


Figure 1: **Routing-equivalence vs. routing-critical edits.** All three DAGs share manipulative ancestors $\{A, B\}$ of Y; D (gray) is non-manipulable and E is a non-ancestor. The schematic assumes matched fitting inputs and unchanged do-priors for the outside edit $B \to E$, giving $\mathcal{G}_1 \sim_{\mathrm{R}} \mathcal{G}_2$. The inside edit $D \to B$ (red) is routing-critical if $d_\mu > \delta_\mu$, as assumed for the displayed relation (Defs. 1–2).

optimization [Branchini et al., 2023]. Graph Agnostic Causal Bayesian Optimization maintains plausible graph models and selects interventions optimistically so that causal discovery is pursued only when it serves reward maximization [Mukherjee et al., 2024]. CBO with Unknown Graphs improves scalability by focusing posterior inference on the direct causal parents of the target variable [Durand et al., 2025]. While these methods weaken the need for expert-specified graphs, a remaining limitation is that graph relevance is still expressed through graph hypotheses, graph posteriors, or structurally restricted target-local summaries. This results in an optimization objective that is too tightly coupled with causal discovery, often demanding significant sample budgets to resolve structural uncertainties that might be irrelevant to the downstream choices of the optimizer.

Drawing insights from causal-bandit research, partial structural information has been shown to suffice for no-regret learning even under hidden confounders [Elahi et al., 2024]. A similar principle applies to our fixed-budget Bayesian optimization setting. What matters to the optimizer is not the exact recovery of the entire graph topology, but rather the optimizer-facing routing objects induced by the graph, such as the runtime ES family and the graph-derived *do*-prior objects. This insight leads to the notions of *routing-equivalence* and *routing-critical edits*. At a fixed update, two graphs are routing-equivalent if they induce the same runtime ES family and the same graph-derived prior objects. A local edit is routing-critical when it changes the ES family or moves the prior mean beyond a specified tolerance. As Figure 1 illustrates, an edit can change the DAG without changing its routing inputs; at matched state and coupled fitting/selector randomness, such an edit preserves the next intervention. An edit within the target-relevant ancestral structure can instead alter admissible ESs or graph-derived priors. Therefore, in latent-confounded settings, the maintained graph should be interpreted as an observed-variable routing proxy rather than a strictly identifiable causal DAG.

Based on this observation, we reframe unknown-graph CBO as *causal belief routing* rather than full causal-graph recovery. In this paper, we propose AGENTCBO, a guarded graph-belief routing framework that treats large language model (LLM) causal outputs as structured but uncertain causal priors. AGENTCBO does not treat the LLM as an autonomous optimizer or an unrestricted causal discovery oracle. Instead, it enforces a strict separation of roles where language models provide structured but uncertain causal beliefs while the BO loop manages intervention selection, surrogate updating, and acquisition. The graph-update stack follows a guarded *Propose, Calibrate, Repair, and Commit* architecture. The LANGUAGE-PRIOR AGENT elicits pairwise causal beliefs and combines them with constraint-based discovery to obtain a Peter-Clark(PC) Algorithm[Spirtes et al., 2000]-anchored point DAG, a protected backbone, and residual uncertain pairs. The CALIBRATION AGENT and RESIDUAL REPAIR AGENT then operate only on residual structures through score-gated refinement and bounded local edits. Candidate updates are accepted through a deterministic COMMIT GATE based on confidence and inexpensive routing-proxy checks. When a graph is committed, runtime ESs and graph-derived *do*-prior objects are recomputed, and archived interventions are re-injected whenever compatible ESs reappear. Interventional outcomes update the per-ES BO surrogate and archive instead of being fed into the LLM-based proposal mechanism.

In summary, our contributions are as follows. (1) We propose AGENTCBO, an LLM-assisted CBO framework whose *Propose, Calibrate, Repair, and Commit* stack admits graph beliefs through PC anchoring, score-gated calibration, bounded repair, and a deterministic commit gate, while

retaining CBO's intervention authority. (2) We formalize *causal belief routing* through the ES and do-prior interface, giving sufficient conditions for fixed-update decision preservation, a quantitative CEI stability bound, and an acquisition-margin condition for retaining the selected ES. (3) Across ten benchmarks, AGENTCBO is generally competitive with known-graph CBO, with its largest refinement gain on CompleteGraph. Matched PC-only ablations, exploration-set controls, route replay, and a controlled corruption study provide evidence for prior utility, finite-budget routing effects, and update-time protection. ALARM evaluates semantic-prior utility at 37 nodes, while ordered ER studies demonstrate operation and quantify computational cost through 50 nodes under a supplied ordering prior.

2 Background and Problem Statement

2.1 Structural Causal Models and Interventions

A system is described by a structural causal model (SCM) $\mathcal{M} = \langle \mathcal{G}, \mathbf{V}, \mathbf{F}, P(\boldsymbol{\omega}) \rangle$, where $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ is a directed acyclic graph (DAG), $\mathbf{V} = \{X_0, \ldots, X_{m-1}, Y\}$ is a set of endogenous observed random variables with target variable $Y = X_m$, $\mathbf{F} = \{f_i\}_{i=0}^{m}$ is a set of unknown structural equations, and $P(\boldsymbol{\omega})$ is a joint distribution over noise variables. Each variable is generated according to $X_i = f_i(\mathbf{Z}_i) + \omega_i$, where $\mathbf{Z}_i = \{X_j : j \in \mathrm{pa}(i)\}$ denotes the parents of X_i in $\mathcal{G}$ and ω_i is zero-mean noise. The set of *manipulative variables* $\mathbf{M} \subseteq \mathbf{V} \setminus \{Y\}$, on which interventions are possible, is distinguished from the set of non-manipulative variables $\mathbf{C} = \mathbf{V} \setminus \mathbf{M} \setminus \{Y\}$. A *hard intervention* $\mathrm{do}(\mathbf{X}_S = \mathbf{x}_S)$ on a subset $S \subseteq \mathbf{M}$ removes all incoming edges to the variables in S and fixes their values to $\mathbf{x}_S$, inducing the interventional distribution $P(Y \mid \mathrm{do}(\mathbf{X}_S = \mathbf{x}_S))$. The *causal effect* of this intervention on the target is $\mu_S(\mathbf{x}_S) = \mathbb{E}[Y \mid \mathrm{do}(\mathbf{X}_S = \mathbf{x}_S)]$, which can be computed from the SCM when the relevant interventional distribution is identifiable, for example via truncated factorization or do-calculus adjustment formulas.

We allow the data-generating SCM to contain latent variables $\mathbf{U}$, in which case the full causal graph is a DAG over $\mathbf{V} \cup \mathbf{U}$ and the noise-independence assumption $P(\boldsymbol{\omega})$ applies to the joint $(\mathbf{V}, \mathbf{U})$ system; after marginalizing $\mathbf{U}$, observed residuals on $\mathbf{V}$ may be statistically dependent and the observed-variable graph is in general a latent projection rather than an ordinary DAG. AGENTCBO maintains an observed-variable graph proxy $\hat{\mathcal{G}}$ over $\mathbf{V}$, used to construct the routing channels (exploration sets and do-priors) of Section 4.2: in fully observed benchmarks $\mathbf{U} = \emptyset$ and $\hat{\mathcal{G}}$ targets the true DAG, while in latent-confounded benchmarks such as COMPLETEGRAPH (Appendix B), $\hat{\mathcal{G}}$ functions as a routing proxy used by the optimizer rather than an identifiable causal graph.

2.2 Causal Bayesian Optimization

CBO aims to minimize the causal effect by jointly selecting the variables for intervention and the corresponding values to assign: $(S^\star, \mathbf{x}^\star_{S^\star}) = \arg\min_{S \in \mathcal{ES},\, \mathbf{x}_S \in \mathcal{D}(S)} \mu_S(\mathbf{x}_S)$, where $\mathcal{ES}$ denotes the exploration-set (ES) family, a collection of causally motivated subsets of the manipulative variables, and $\mathcal{D}(S)$ is the interventional domain for S. In standard CBO, the ES family is derived from the causal graph. Each $S \in \mathcal{ES}$ is a subset of the manipulative ancestors of Y in $\mathcal{G}$, and the possibly-optimal minimal intervention sets (POMIS) provide the theoretically minimal collection [Lee and Bareinboim, 2018]. CBO maintains a separate Gaussian process (GP) surrogate $\hat{f}_S \sim \mathcal{GP}(\mu_S^{\mathrm{prior}}, k_S)$ for each ES $S \in \mathcal{ES}$. The prior mean μ_S^{prior} is informed by do-calculus propagation through a network of per-node conditional GPs fitted to observational data. At each iteration, CBO selects the ES and intervention values that maximize the cost-normalized causal expected improvement (CEI):

$$(S_t, \mathbf{x}_t) = \underset{S \in \mathcal{ES},\, \mathbf{x}_S \in \mathcal{D}(S)}{\arg\max} \frac{\mathrm{EI}_S(\mathbf{x}_S)}{\mathrm{cost}(S, \mathbf{x}_S)}. \tag{1}$$

Between interventions, an ϵ-greedy schedule is used to collect additional observational data, update the conditional GP network, and refine the do-calculus prior.

Standard known-graph setting. The standard CBO framework assumes that the causal DAG $\mathcal{G}$ is known. This assumption is also used by model-based extensions, such as model-based CBO (MCBO) and adversarial CBO. This work relaxes the known-graph setting. Rather than treating the resulting problem solely as a sequential task of causal discovery followed by optimization, the

causal graph $\mathcal{G}$ is considered as the input to a deterministic mapping pipeline for the optimizer: $\mathcal{G} \mapsto \mathcal{ES}(\mathcal{G}) \mapsto \{\mu_S^{\text{prior}}(\mathcal{G}), \sigma_S^{\text{prior}}(\mathcal{G})\}_{S \in \mathcal{ES}(\mathcal{G})} \mapsto \Pi_{\mathcal{G}}$, where $\Pi_{\mathcal{G}}$ denotes the CEI-driven intervention policy of Eq. (1). In this routing-policy framework, the optimizer interacts with the graph through the induced ESs and the do-calculus prior objects used by the surrogate, rather than through the full topology directly. As a result, two graphs that yield identical outputs along this mapping are indistinguishable to the BO policy via these routing channels, even if they differ in structural details that do not affect target-relevant intervention routes. This perspective motivates the evaluation of estimated graphs by the intervention routes they induce, in addition to structural recovery metrics.

3 Related Work

3.1 CBO under Unknown Graphs

CBO [Aglietti et al., 2020] reduces intervention costs by leveraging a known causal graph to define exploration sets and do-calculus priors. Subsequent extensions have enhanced CBO in adversarial [Sussex et al., 2024], constrained [Aglietti et al., 2023], and multi-objective [Bhatija et al., 2025] contexts; however, most approaches assume the causal graph is known in advance. Relaxing this assumption remains challenging. Existing unknown-graph CBO methods differ along four axes: the structural object maintained, the evidence updating it, the update schedule, and how it couples to the BO loop (Table 1). CEO [Branchini et al., 2023] maintains a posterior over DAGs updated by joint observational and interventional likelihood, with information-theoretic acquisition. GACBO [Mukherjee et al., 2024] maintains plausible (graph, function, action) triples updated by Bayesian scores and GP confidence sets, with UCB selection. CBO-U [Durand et al., 2025] maintains a posterior over the target's direct parents updated by parent-set CI tests, with a parent-aware surrogate. POMIS characterizes possibly-optimal minimal intervention sets for a given graph [Lee and Bareinboim, 2018], while Elahi et al. [2024] identify partial structure sufficient for no-regret causal-bandit learning. Park et al. [2025] extend intervention-set characterization to Markov-equivalence classes represented by partial ancestral graphs. CEO also illustrates optimization-equivalent graphs with equal causal-effect prior means and covariances [Branchini et al., 2023]. Our analysis jointly tracks the graph-dependent RuntimeES and rebuilt predictive interface, establishing conditional fixed-update selector preservation and CEI stability.

Table 1: Positioning of AGENTCBO relative to unknown-graph CBO baselines along four dimensions.

Method	Structural scope	Evidence	Belief update	BO coupling
Dynamic PC-only	DAG	CI tests	Periodic	Guarded ESs and priors
CEO [Branchini et al., 2023]	DAG posterior	Obs+int likelihood	Online	Entropy search
GACBO [Mukherjee et al., 2024]	Plausible DAGs	Bayes scores + GP CS	Online	UCB over (G, f, a)
CBO-U [Durand et al., 2025]	Parents of Y	Parent-set CI	Online	Parent-aware surrogate
AGENTCBO (ours)	Target-routed DAG	CI + LLM prior, guarded	Periodic, gated	Routing-aware ESs and priors

3.2 LLMs in Causal Discovery and Scientific Optimization

Recent evaluations reveal that while LLMs excel at pairwise causal judgments, they struggle with full graph recovery and often conflate correlation with causation [Kiciman et al., 2024, Jin et al., 2024, Chen et al., 2024]. A robust mitigation strategy integrates LLM knowledge as soft priors into traditional discovery algorithms, allowing statistical data to override inconsistent LLM beliefs [Long et al., 2023]. LLM-assisted BO uses language models for surrogate modeling and candidate generation [Liu et al., 2024, Cissé et al., 2025, Agarwal et al., 2025], while other approaches use them as direct optimizers [Yang et al., 2024, Roohani et al., 2025]. LGBO [Yuan et al., 2026] incorporates iterative LLM preferences through region-lifted GP mean shifts. LABO [Chen et al., 2026] combines LLM predictions and experimental observations in a multi-fidelity surrogate with discrepancy-based evaluation gating. AGENTCBO uses LLM guidance in graph construction and guarded commitment, linking the committed graph to RuntimeES and do-priors while CBO selects interventions.

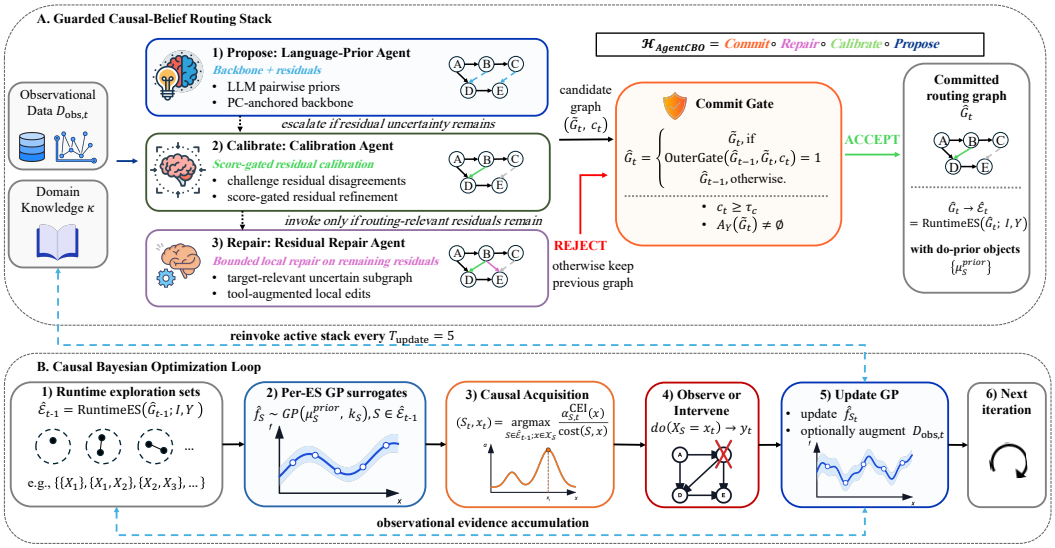


Figure 2: **AGENTCBO guarded routing pipeline.** The LANGUAGE-PRIOR AGENT proposes a causal backbone and routing-relevant residual pairs, the CALIBRATION AGENT imports score-improving statistical calibrations, and the RESIDUAL REPAIR AGENT applies bounded local repair. The deterministic COMMIT GATE compares the stacked candidate $\tilde{G}_t$ with the previous committed graph $\hat{G}_{t-1}$ and commits the graph used by CBO. The committed graph induces RuntimeES and do-priors for cost-normalized CEI. The outer loop either observes a fresh batch or executes an intervention; interventions update the surrogate and archive, not graph discovery by default.

4 Causal Belief Routing

4.1 Problem Setting

Given an observational dataset $\mathcal{D}_{\text{obs}} = \{\mathbf{v}^{(i)}\}_{i=1}^{n_{\text{obs}}}$ drawn from the observed marginal of an unknown SCM $\mathcal{M}$ whose full graph $\mathcal{G}^\star$ may include latent variables (Section 2), with manipulative subset $\mathbf{M} \subseteq \mathbf{V}$, AGENTCBO maintains an estimated graph $\hat{\mathcal{G}}_t$ at each round t and selects interventions with CEI as defined in Eq. (1). In this section, we write $\mathcal{I} := \mathbf{M}$ for the intervention-eligible variables to avoid overloading $\mathcal{M}$. Let $\mathcal{A}_Y(\mathcal{G}) := \text{anc}_{\mathcal{G}}(Y) \cap \mathcal{I}$ denote the manipulative ancestors of Y. While the ideal CBO construction utilizes $\mathcal{ES}_{\text{ideal}}(\mathcal{G}) := 2^{\mathcal{A}_Y(\mathcal{G})} \setminus \{\emptyset\}$, the implemented optimizer instead employs a runtime constructor

$$\widehat{\mathcal{E}}(\mathcal{G}) = \text{RuntimeES}(\mathcal{G}; \mathcal{I}, Y). \tag{2}$$

This constructor retains singleton manipulative interventions, truncates excessively large families, and defaults to a conservative strategy when no manipulative ancestor is identified. During graph-update rounds, the active stack outputs a candidate graph $\tilde{\mathcal{G}}_t$, a confidence scalar c_t, and metadata for the outer gate. Subsequently, the outer loop integrates the committed graph with the conditional GP network fitted on $\mathcal{D}_{\text{obs}}$ to compute the do-calculus prior objects required by CEI.

4.2 Routing-Equivalence and Routing-Critical Edits

The routing-policy perspective described in Section 2 treats the graph as input to the runtime ES constructor, the graph-derived do-prior objects, and ultimately the CEI policy. Consequently, the appropriate unit of analysis is not an individual edge, but rather the equivalence class of graphs that yield identical optimizer-facing routing inputs.

Definition 1 (Routing-equivalence, fixed update)**.** *At a fixed CBO update with matched non-routing inputs, two graphs $\mathcal{G}_1, \mathcal{G}_2$ are **routing-equivalent**, written $\mathcal{G}_1 \sim_{\text{R}} \mathcal{G}_2$, if they induce the same labeled runtime ES family, $\widehat{\mathcal{E}}(\mathcal{G}_1) = \widehat{\mathcal{E}}(\mathcal{G}_2)$, and pointwise-equal graph-derived do-prior objects on every admissible training or query input used by the surrogate builder.*

At matched inputs, labeled graph equality implies routing-equivalence, whereas Markov equivalence is generally not comparable to it. Markov-equivalent graphs can induce different intervention routes, and routing-equivalent graphs can differ away from those routes.

Definition 2 (Routing-critical edit)**.** *For a local graph edit Δ, let $\mathcal{G} \oplus \Delta$ denote the edited graph. Given a tolerance $\delta_\mu \geq 0$, the edit is **routing-critical** if it changes the runtime ES family, $\widehat{\mathcal{E}}(\mathcal{G}) \neq \widehat{\mathcal{E}}(\mathcal{G} \oplus \Delta)$, or if the do-prior mean distance satisfies $d_\mu(\mathcal{G}, \mathcal{G} \oplus \Delta) > \delta_\mu$.*

The outer gate described below does not compute this definition exactly. Instead, it uses manipulative target ancestry and normalized structural Hamming distance (nSHD) on the ancestor subgraph as computationally efficient proxies for routing-criticality. The complete distance-based definition and the gate predicate are provided in Appendix D.

Fixed-update guarantees. Hold fixed the data, archive, domains, costs, and fitting/selection rules *before* rebuilding route-conditioned surrogates. All graph dependence of the common builder and selector must enter through the labeled ES family and graph-derived prior objects. Fitted predictions are outputs, not fixed inputs. Appendix G gives the full assumptions and proofs.

Lemma 1 (Exact selector sufficiency)**.** *Under the matched-state conditions of Appendix G.1, routing-equivalent graphs give the same conditional next-intervention distribution when the surrogate builder and numerical selector use common graph-independent randomness and tie-breaking.*

Proposition 1 (Approximate CEI stability)**.** *Compare two rebuilt Gaussian predictors on the same finite nonempty labeled ES family and nonempty domains, with the same cost $c(a) \geq c_{\min} > 0$ for action $a = (S, \mathbf{x})$. Write $g = b - m$ and $\widetilde{g} = \widetilde{b} - \widetilde{m}$ for minimization improvement means, where b is the EI incumbent, and $s, \widetilde{s} \geq 0$ for predictive standard deviations. Let $\alpha(a) = \mathbb{E}[(g(a) + s(a)U)_+]/c(a)$ for $U \sim \mathcal{N}(0,1)$, and define $\widetilde{\alpha}$ analogously. If $\|g - \widetilde{g}\|_\infty \leq \epsilon_g$ and $\|s - \widetilde{s}\|_\infty \leq \epsilon_s$, with finite bounds, then*

$$\|\alpha - \widetilde{\alpha}\|_\infty \leq \eta := \frac{\epsilon_g + \epsilon_s/\sqrt{2\pi}}{c_{\min}}. \tag{3}$$

If $\widetilde{\alpha}(\widetilde{a}) \geq \sup_a \widetilde{\alpha}(a) - \zeta$ for finite $\zeta \geq 0$, then $\sup_a \alpha(a) - \alpha(\widetilde{a}) \leq 2\eta + \zeta$. If the reference-optimal ES has a peak-acquisition margin $\Gamma > 2\eta + \zeta$, its identity is preserved (Corollary 1).

The proof uses the Gaussian-EI sensitivities $|\partial_g \mathrm{EI}| \leq 1$ and $|\partial_s \mathrm{EI}| \leq 1/\sqrt{2\pi}$, followed by an approximate-maximizer comparison. These results link the joint ES–surrogate interface to intervention selection without requiring full graph recovery. The guarantees apply at a matched fixed update and require control of both predictive means and uncertainties. The implemented proxy gate is evaluated separately in Appendix C.4.

4.3 AGENTCBO: A Guarded LLM-Agent Stack

Setup. AGENTCBO instantiates the routing view with a guarded graph-update stack. At update rounds, the LANGUAGE-PRIOR AGENT proposes a point DAG, frozen backbone, and residual candidate pairs; the CALIBRATION AGENT and RESIDUAL REPAIR AGENT may modify only those residual pairs, and only when the projected edit improves the internal structural score. The deterministic COMMIT GATE then decides whether the candidate graph replaces the graph currently routing CBO. The reported variants activate progressively more stages: AGENTCBO-PRIOR uses proposal only, AGENTCBO-CALIB adds calibration, and AGENTCBO-FULL adds residual repair. Interventional observations update the BO surrogate and archive, not the LLM-based graph proposal. The full LLM input–output contract for each agent is specified in Appendix F. We use "belief" operationally, not as a Bayesian posterior over DAGs: the router maintains a committed graph $\hat{\mathcal{G}}_t$, a Prior-Agent proposal $\mathcal{G}_t^{\mathrm{LP}}$, a frozen backbone $\mathcal{B}_t$, residual pairs $\mathcal{R}_t$, an internal structural score $s(\mathcal{G})$, the runtime ES family $\widehat{\mathcal{E}}_t$, and a cross-graph archive $\mathcal{H}_t$.

Propose stage. The LANGUAGE-PRIOR AGENT combines pairwise LLM causal beliefs with PC anchoring, CPDAG-to-DAG extension, guarded edge supplementation, and confidence scoring. It also constructs a frozen backbone and residual pairs:

$$\left(\mathcal{G}_t^{\mathrm{LP}}, \mathcal{B}_t, \mathcal{R}_t, c_t^{\mathrm{LP}}\right) = \mathrm{Propose}_{\mathrm{LP}}(\mathcal{D}_{\mathrm{obs},t}, \kappa), \qquad \mathcal{R}_t = \left(\mathcal{R}_t^{\mathrm{low}} \cup \mathcal{R}_t^{\mathrm{conflict}} \cup \mathcal{R}_t^{\mathrm{target}}\right) \setminus \mathcal{B}_t. \tag{4}$$

The complete LP details are given in Algorithm 2 in Appendix D.

Refine stages. Starting from $\mathcal{G}_t^0 = \mathcal{G}_t^{\mathrm{LP}}$, the calibration and repair stages share one guarded refinement interface. For stages $k \in \{1, 2\}$, let $a_1 = \mathrm{CC}$ and $a_2 = \mathrm{RA}$. The stage proposes a challenger graph $\mathcal{G}_t^{a_k}$, and the stack imports only edits that lie in the active residual set, preserve the frozen backbone, obey the stage edit budget, and keep the graph acyclic:

$$\bar{\mathcal{G}}_t^k = \mathrm{Project}\big(\mathcal{G}_t^{a_k};\, \mathcal{G}_t^{k-1}, \mathcal{R}_t^k, \mathcal{B}_t, m_k\big)\,. \tag{5}$$

The projected challenger is accepted only when it improves the internal structural score:

$$\mathcal{G}_t^k = \begin{cases} \bar{\mathcal{G}}_t^k, & s(\bar{\mathcal{G}}_t^k) < s(\mathcal{G}_t^{k-1}) - \Delta_k, \\ \mathcal{G}_t^{k-1}, & \text{otherwise.} \end{cases} \tag{6}$$

Here $m_1 = \infty$ for the CALIBRATION AGENT and $m_2 = m_{\max}$ for the RESIDUAL REPAIR AGENT. We use $\Delta_1 = \Delta_2 = 0$ and $m_{\max} = 5$; acceptance-rate diagnostics and threshold rationale are reported in Appendix E. If no projected edit improves the score, the stage is a no-op, and the stack returns the single candidate $\tilde{\mathcal{G}}_t = \mathcal{G}_t^2$ to the COMMIT GATE.

Score function. The internal score is a statistical structural score, with lower values preferred:

$$s(\mathcal{G}) = \mathrm{NLL}_{\mathrm{holdout}}(\mathcal{G}) + \lambda_{\mathrm{CI}} r_{\mathrm{CI}}(\mathcal{G}) + \lambda_{\mathrm{sp}} \mathrm{SP}(\mathcal{G}). \tag{7}$$

The three terms are holdout structural-equation negative log-likelihood, a conditional-independence (CI)-consistency penalty over d-separation implications, and a scaled edge-count sparsity penalty.

Guarded design. The LANGUAGE-PRIOR AGENT's proposal is **protected by default**: later stages refine only residual pairs through the guarded interface (Eqs. (5)–(6)), with the proposal preserved when no projected edit improves the score. The full stack therefore reduces to AGENTCBO-PRIOR when the prior graph is sufficient, to AGENTCBO-CALIB when no repair edit is justified, or commits a repaired graph only when both the internal score and the COMMIT GATE support the change.

4.4 Outer Loop, Commit Gate, and Cross-Graph Archival

The active discovery stack is invoked inside a shared outer loop that interleaves observation, intervention, and graph rediscovery. At each iteration, the loop either collects a fresh observational batch or executes an intervention according to an ϵ-greedy observe-versus-intervene schedule (formalized in Algorithm 1). Observational rounds augment $\mathcal{D}_{\mathrm{obs}}$, refit the per-node conditional GP network, and refresh the do-priors μ_S^{prior}; intervention rounds select $(S_t, \mathbf{x}_t)$ by cost-normalized CEI and update the corresponding per-ES GP. Every $T_{\mathrm{update}} = 5$ iterations the selected AGENTCBO-PRIOR, AGENTCBO-CALIB, or AGENTCBO-FULL stack is re-invoked using the current observational data; in the calibrated and full variants, the CALIBRATION AGENT and RESIDUAL REPAIR AGENT are conditional refinement stages and a no-op outcome is part of the algorithm when calibration finds no accepted residual edit or repair finds no routing-critical correction worth committing.

Commit Gate. The stack returns one candidate $\tilde{\mathcal{G}}_t$; the deterministic COMMIT GATE decides whether this candidate replaces the currently committed graph:

$$\hat{\mathcal{G}}_t = \begin{cases} \tilde{\mathcal{G}}_t, & \mathrm{OuterGate}(\hat{\mathcal{G}}_{t-1}, \tilde{\mathcal{G}}_t, c_t) = 1, \\ \hat{\mathcal{G}}_{t-1}, & \text{otherwise.} \end{cases} \tag{8}$$

The COMMIT GATE is a cheap proxy for routing-criticality rather than a direct RuntimeES-distance optimizer: it requires sufficient confidence, rejects candidates with no manipulative ancestor of Y, and raises the confidence bar when the target-ancestor subgraph changes substantially. The full threshold predicate is given in Appendix D, Eq. (10).

Cross-graph data archival. A practical bottleneck in sequential discovery-optimization is the stranding of interventional data when the accepted point graph changes. When COMMIT GATE triggers a graph change, AGENTCBO archives all observed interventional data $\{(S, \mathbf{x}_S, y)\}$ keyed by their ES signature S. Upon any subsequent graph change that re-admits S, these archived tuples are re-injected into the per-ES GP. This is specific to our point-graph switching loop: continuous-posterior methods update structural beliefs directly rather than requiring explicit ES re-admission.

Surrogate and acquisition. For each active $S \in \widehat{\mathcal{E}}_t$, we maintain a GP $\hat{f}_S$ with the CausalRBF kernel, in which the do-calculus variance of the prior mean contributes an additive term that inflates the uncertainty induced by prior propagation under the accepted graph. The prior mean μ_S^{prior} is computed by the outer loop from $\hat{\mathcal{G}}$ and the per-node conditional GPs fitted on $\mathcal{D}_{\text{obs}}$. We follow the standard CBO ϵ-greedy observe-versus-intervene schedule, and intervention rounds use

$$(S_t, \mathbf{x}_t) = \underset{S \in \widehat{\mathcal{E}}(\hat{\mathcal{G}}_{t-1}),\, \mathbf{x} \in \mathcal{D}(S)}{\arg\max} \frac{\text{EI}_{S,t}(\mathbf{x})}{\text{cost}(S, \mathbf{x})}. \tag{9}$$

Across all reported benchmarks we set $\text{cost}(S, \mathbf{x}) = |S|$, penalizing the cardinality of the intervention set; this matches the cardinality-cost convention used by standard CBO [Aglietti et al., 2020].

Algorithm 1 AgentCBO outer loop with active graph-update stack.

Require: Budget T; rediscovery period T_{update}; observational data $\mathcal{D}_{\text{obs}}$; domain context κ.
Ensure: Best routed intervention $(S^\star, \mathbf{x}^\star)$ with target value $Y^\star$.

$(\hat{\mathcal{G}}_0, \mathcal{B}_0, \mathcal{R}_0, c_0) \leftarrow \text{StackUpdate}(\mathcal{D}_{\text{obs}}, \mathbf{V}, \kappa)$ ▷ AGENTCBO-PRIOR / AGENTCBO-CALIB / AGENTCBO-FULL
$\widehat{\mathcal{E}}_0 \leftarrow \text{RuntimeES}(\hat{\mathcal{G}}_0; \mathcal{I}, Y)$; fit per-ES GPs $\{\hat{f}_S\}$ with CausalRBF kernel.
for $t = 1, \ldots, T$ **do**
  $(\hat{\mathcal{G}}_t, \widehat{\mathcal{E}}_t) \leftarrow (\hat{\mathcal{G}}_{t-1}, \widehat{\mathcal{E}}_{t-1})$ ▷ carry forward
  $b_t \sim \text{Bernoulli}(\epsilon_t)$ ▷ ϵ-greedy schedule
  if $b_t = 1$ **then** ▷ observe
    Augment $\mathcal{D}_{\text{obs}}$; refit per-node GPs and do-priors $\{\mu_S^{\text{prior}}\}$.
  else ▷ intervene
    $(S_t, \mathbf{x}_t) \leftarrow \arg\max_{S \in \widehat{\mathcal{E}}_{t-1},\, \mathbf{x} \in \mathcal{D}(S)} \text{EI}_{S,t}(\mathbf{x}) / \text{cost}(S, \mathbf{x})$ ▷ Eq. (9)
    $y_t \leftarrow \text{do}(\mathbf{x}_t)$; update $\hat{f}_{S_t}$.
  end if
  if $t \bmod T_{\text{update}} = 0$ **then** ▷ rediscover
    $(\tilde{\mathcal{G}}_t, \mathcal{B}_t, \mathcal{R}_t, c_t) \leftarrow \text{StackUpdate}(\mathcal{D}_{\text{obs}}, \mathbf{V}, \kappa)$
    if $\text{OuterGate}(\hat{\mathcal{G}}_{t-1}, \tilde{\mathcal{G}}_t, c_t) = 1$ **then** ▷ Eq. (10)
      $\hat{\mathcal{G}}_t \leftarrow \tilde{\mathcal{G}}_t$; $\widehat{\mathcal{E}}_t \leftarrow \text{RuntimeES}(\hat{\mathcal{G}}_t; \mathcal{I}, Y)$; re-fit $\{\hat{f}_S\}$ on archived data.
    end if
  end if
end for
return $(S^\star, \mathbf{x}^\star, Y^\star)$ from $\arg\min_{t \le T,\, b_t = 0} y_t$.

5 Experiments

5.1 Experimental Setup

Benchmarks and baselines. The evaluation is conducted on ten synthetic and semi-synthetic DAGs spanning four categories: small toy graphs (CHAIN-5, TOYGRAPH); application-motivated structures (PSAGRAPH, HIVEPIDEMIOLOGY, CCBOHEALTH); multi-ancestor graphs (PISHCATGRAPH, PREDATORPREY); and the hidden-confounder benchmark (COMPLETEGRAPH), along with standard optimization surfaces (ROSENBROCK, ACKLEY) reformulated as causal problems. The number of observed variables in these graphs ranges from 3 to 11. Appendix B summarizes benchmark details. The evaluation includes comparisons with graph-free references, **BO**, **LLAMBO**, and **BORA**; a known-graph **CBO** reference; and unknown-graph CBO baselines, **CBO-U**, **Graph-Agnostic CBO (GACBO)**, and **Causal Entropy Optimization (CEO)**. Within AgentCBO, AGENTCBO-PRIOR uses only the proposal layer, AGENTCBO-CALIB adds statistical calibration of residual disagreements, and AGENTCBO-FULL adds bounded residual repair. The main matrix reports all three AgentCBO variants; standalone **CC** and **RA** are reported only as separate module diagnostics. Additional diagnostics and the dynamic PC-only ablation are in Appendix C. Our code is available at `https://github.com/Qiyu-Wei/AgentCBO_Official`.

Table 2: **Main result matrix.** Mean ± standard deviation of best objective $Y^\star$ reached within 50 outer-loop steps, aggregated across 20 seeds per benchmark–algorithm pair. All benchmarks are minimization tasks. **Bold** marks the best displayed mean among methods other than the known-graph CBO reference baseline; ties at the displayed precision are bolded together.

	Reference	*CBO baselines*				*LLM-assisted BO*		*AgentCBO variants*		
Benchmark	BO	CBO	CEO	CBO-U	GACBO	LLAMBO	BORA	Prior	Calib	Full
CompleteGraph	-0.01 ± 0.35	-2.48 ± 1.14	-1.43 ± 0.55	-0.57 ± 0.16	-1.23 ± 0.12	-0.40 ± 0.35	-0.73 ± 0.04	-2.50 ± 1.21	$\mathbf{-3.57\pm0.07}$	$\mathbf{-3.57\pm0.07}$
ToyGraph	-1.94 ± 0.22	-2.08 ± 0.15	-2.02 ± 0.09	-1.56 ± 0.32	-1.74 ± 0.36	-1.95 ± 0.12	-1.92 ± 0.16	$\mathbf{-2.11\pm0.17}$	-2.02 ± 0.16	-2.10 ± 0.14
PSAGraph	$+5.40\pm0.09$	$+5.19\pm0.15$	$+5.22\pm0.06$	$\mathbf{+5.15\pm0.02}$	$+5.62\pm0.02$	$+5.34\pm0.01$	$+5.34\pm0.01$	$+5.18\pm0.07$	$+5.19\pm0.08$	$+5.19\pm0.08$
CCBOHealth	$+5.40\pm0.14$	$+5.05\pm0.21$	$+5.36\pm0.11$	$+5.00\pm0.06$	$+5.83\pm0.19$	$+5.02\pm0.12$	$+4.99\pm0.14$	$\mathbf{+4.96\pm0.09}$	$\mathbf{+4.96\pm0.09}$	$\mathbf{+4.96\pm0.09}$
HIVEpidemiology	-1.39 ± 0.04	-1.49 ± 0.01	$\mathbf{-1.50\pm0.02}$	-1.43 ± 0.08	-1.33 ± 0.41	-1.48 ± 0.01	$\mathbf{-1.50\pm0.01}$	$\mathbf{-1.50\pm0.01}$	-1.49 ± 0.01	-1.49 ± 0.01
Chain-5	-1.21 ± 0.09	-1.32 ± 0.00	-1.25 ± 0.07	-1.20 ± 0.13	-1.25 ± 0.01	-1.26 ± 0.07	-1.30 ± 0.02	$\mathbf{-1.32\pm0.01}$	$\mathbf{-1.32\pm0.01}$	$\mathbf{-1.32\pm0.01}$
Rosenbrock	-52.85 ± 16.57	-89.45 ± 8.23	-49.86 ± 31.37	-83.88 ± 2.93	$+3.68\pm0.10$	-90.60 ± 0.49	-86.19 ± 3.52	-90.98 ± 0.05	$\mathbf{-90.99\pm0.02}$	$\mathbf{-90.99\pm0.02}$
Ackley	-6.94 ± 0.36	-7.39 ± 0.02	-6.82 ± 0.41	-6.74 ± 0.54	-5.84 ± 0.04	-7.14 ± 0.21	-7.43 ± 0.02	$\mathbf{-7.51\pm0.01}$	-7.50 ± 0.01	-7.50 ± 0.01
PredatorPrey	-2.28 ± 0.04	-2.21 ± 0.02	-2.31 ± 0.02	-2.07 ± 0.19	-1.82 ± 0.01	-2.29 ± 0.03	-2.28 ± 0.09	-2.31 ± 0.01	$\mathbf{-2.32\pm0.00}$	-2.31 ± 0.01
PISHCATGraph	$+0.34\pm0.05$	$+0.22\pm0.01$	$+65.28\pm0.11$	$+0.22\pm0.04$	$+0.90\pm0.01$	$+0.25\pm0.03$	$+0.26\pm0.02$	$+0.30\pm0.10$	$+0.22\pm0.09$	$\mathbf{+0.21\pm0.09}$

Protocol. Each experimental run begins with $n_{\text{obs}} = 100$ observational samples and a total optimization budget of $T = 50$ outer-loop steps. The result matrix uses 20 random seeds. All LLM-based engines utilize a Qwen3.5-35B-A3B model with vLLM; the serving setups are documented in Appendix D. We report the best objective value $Y^\star$ achieved within the budget, along with the mean and standard deviation across seeds. All main experiments use this fixed budget; systematic sweeps over the intervention budget are left to future work.

5.2 Benchmark Result Matrix and Routing Effects

Figure 3 shows the ten-benchmark trajectories; Table 2 reports their endpoints. The variants are generally competitive with known-graph CBO. Paired tests across benchmark means favor Prior over BO and GACBO after Holm correction across nine comparisons (both adjusted $p = 0.018$; Appendix C.2). Refinement gains vary across the reported benchmark means. Suite-level tests detect no difference between Prior and Calib ($p = 0.84$) or Full ($p = 0.83$); the CompleteGraph controls below examine the largest observed gain.

CompleteGraph mechanism. On the hidden-confounder benchmark, Calib and Full improve the reported endpoint from Prior's -2.50 to -3.57. Separate fixed-graph controls hold cardinality at three sets: the $\{B\}, \{D\}, \{B, D\}$ family reaches $y \leq -3.0$ on 14/20 seeds, versus 4/20 for random pair-induced families and 0/20 for either alternative pair. In one successful traced run, a step-15 commit contracts RuntimeES from seven to four sets, increasing $\{B, D\}$ selection from $1/11$ to $18/27$ intervention decisions. Replaying this schedule under the known graph recovers about 84% of that seed's endpoint gap. These complementary controls connect the gain to finite-budget concentration and retained-set composition (Appendix B.1).

5.3 Module Diagnostics and Stage Ablations

Module diagnostics. The matched no-LLM arm is *dynamic PC-only AgentCBO*. It replaces only the graph-proposal engine while retaining the same guarded CBO pipeline and settings. Prior improves on eight of ten benchmarks, ties on CHAIN-5, and trails by 0.02 on PISHCATGRAPH (Appendix C.1). Standalone CC/RA diagnostics are separate from the stacked Calib/Full ablations (Table 3). The standalone RA engine uses about $34\times$ LP's tokens (Table 15), measured independently of the stacked variants.

Routing channels and the implemented guard. Of 323 graph pairs with identical RuntimeES, 141 (44%) exceed a fitted prior-mean-distance threshold of 0.05, demonstrating that the prior channel carries information beyond ES equality (Appendix C.3). A separate PSAGRAPH stress test starts from a valid incumbent and disables coverage rescue in all arms. The prescribed corrupted update causes zero endpoint degradation with the gate active; bypassing the gate worsens all ten matched endpoints, with mean paired degradation $+0.1387$ (95% CI $[+0.0086, +0.2839]$). This tests update-time protection under the specified corruption, alongside the conditional interface theory (Appendix C.4).

Semantic robustness. LP-Anon replaces variable names with opaque tokens V_i and omits domain descriptions; LP-Shuffled permutes the variable-to-description mapping. Both match or improve

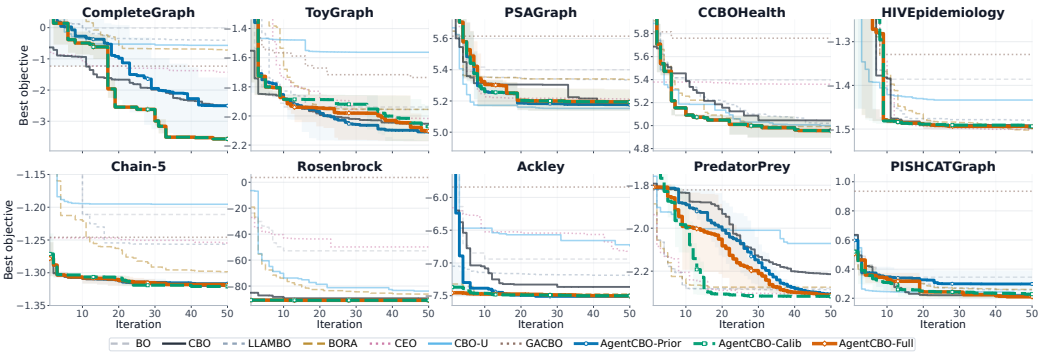


Figure 3: **Best-objective trajectories across the benchmark suite.** Per-benchmark best-objective curves vs. iteration ($T = 50$), aggregated across 20 seeds with shaded variance bands.

Table 3: **Module and semantic-robustness diagnostics.** Mean $\pm$ standard deviation of best $Y^\star$ over 20 seeds; **Bold** marks the best mean among executable-engine diagnostics.

	Executable-engine diagnostics			Semantic-robustness ablations		Component ablation
Benchmark	LP	CC	RA	LP-Anon	LP-Shuffled	Dynamic PC-only
CompleteGraph	$\mathbf{-2.50 \pm 1.21}$	-1.71 ± 1.43	-1.62 ± 1.32	-1.97 ± 1.59	-1.97 ± 1.59	-2.13 ± 1.45
ToyGraph	$\mathbf{-2.11 \pm 0.17}$	-2.07 ± 0.14	-2.10 ± 0.12	-2.10 ± 0.12	-2.10 ± 0.12	-2.02 ± 0.16
PSAGraph	$\mathbf{+5.18 \pm 0.07}$	$+5.35 \pm 0.13$	$+5.36 \pm 0.13$	$+5.31 \pm 0.12$	$+5.24 \pm 0.09$	$+5.35 \pm 0.13$
CCBOHealth	$+4.96 \pm 0.09$	$\mathbf{+4.94 \pm 0.08}$	$+4.96 \pm 0.09$	$+4.92 \pm 0.06$	$+4.92 \pm 0.06$	$+4.98 \pm 0.09$
HIVEpidemiology	$\mathbf{-1.50 \pm 0.01}$	-1.49 ± 0.01	-1.49 ± 0.01	-1.49 ± 0.01	-1.49 ± 0.01	-1.48 ± 0.01
Chain-5	$\mathbf{-1.32 \pm 0.01}$	$\mathbf{-1.32 \pm 0.01}$	$\mathbf{-1.32 \pm 0.01}$	-1.32 ± 0.01	-1.32 ± 0.01	-1.32 ± 0.01
Rosenbrock	-90.98 ± 0.05	$\mathbf{-91.00 \pm 0.00}$	-83.10 ± 24.97	-91.00 ± 0.01	-91.00 ± 0.01	-89.80 ± 0.02
Ackley	$\mathbf{-7.51 \pm 0.01}$	-7.50 ± 0.01	-6.76 ± 0.76	-7.51 ± 0.01	-7.51 ± 0.01	-7.49 ± 0.01
PredatorPrey	$\mathbf{-2.31 \pm 0.01}$	-2.21 ± 0.22	-2.17 ± 0.23	-2.25 ± 0.14	-2.20 ± 0.19	-2.17 ± 0.22
PISHCATGraph	$+0.30 \pm 0.10$	$+0.21 \pm 0.00$	$\mathbf{+0.20 \pm 0.01}$	$+0.20 \pm 0.00$	$+0.23 \pm 0.07$	$+0.28 \pm 0.01$

Prior on five of ten benchmarks (Table 3), showing that performance is not driven solely by lexical names, while degradation on selected tasks supports a role for semantic cues. Appendix F details the semantic and observational inputs available to each agent.

5.4 Larger-Graph Evaluation

On a 37-node ALARM-based synthetic SEM, the prior improves over matched PC-only on all ten seeds when coverage rescue is disabled for component isolation (two-sided sign-test $p = 0.002$); enabling the default rescue makes the arms tie. On generated ER graphs with 10, 20, and 50 nodes, a disclosed ordering prior yields favorable paired normalized-gap point estimates at 20 and 50 nodes, but the single-size confidence intervals include or touch zero. Median optimization time is lower than PC-only at both larger sizes. These studies evaluate semantic-prior utility on a public topology and operation under supplied structural information, respectively (Appendix C.5).

6 Conclusion

AGENTCBO integrates uncertain LLM causal priors into CBO through guarded graph updates while leaving intervention selection to the optimizer. Fixed-update selector preservation and CEI stability explain the joint role of RuntimeES and do-priors. Matched ablations, family controls, single-run route replay, and a corruption study provide complementary evidence for prior utility, finite-budget routing effects, and update-time protection. Larger-graph studies demonstrate operation through 50 nodes under supplied structural information. Together, these results support guarded graph-belief updates as a practical interface between LLM priors and CBO when no trusted graph is available.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their insightful comments and constructive suggestions. We also acknowledge the University of Manchester's CSF3 facility for providing the GPU resources used in this work.

References

Dhruv Agarwal, Manoj Ghuhan Arivazhagan, Rajarshi Das, Sandesh Swamy, Sopan Khosla, and Rashmi Gangadharaiah. Searching for optimal solutions with LLMs via Bayesian optimization. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Virginia Aglietti, Xiaoyu Lu, Andrei Paleyes, and Javier González. Causal Bayesian optimization. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 3155–3164. PMLR, 2020.

Virginia Aglietti, Neil Dhir, Javier González, and Theodoros Damoulas. Dynamic causal Bayesian optimization. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 10549–10560, 2021.

Virginia Aglietti, Alan Malek, Ira Ktena, and Silvia Chiappa. Constrained causal Bayesian optimization. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 304–321. PMLR, 2023.

Ingo A. Beinlich, Henri Jacques Suermondt, R. Martin Chavez, and Gregory F. Cooper. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *AIME 89, Second European Conference on Artificial Intelligence in Medicine, London, UK, August 29th-31st 1989. Proceedings*, volume 38 of *Lecture Notes in Medical Informatics*, pages 247–256. Springer, 1989.

Shriya Bhatija, Paul-David Joshua Zuercher, Jakob Thumm, and Thomas Bohné. Multi-objective causal Bayesian optimization. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025.

Nicola Branchini, Virginia Aglietti, Neil Dhir, and Theodoros Damoulas. Causal entropy optimization. In *International Conference on Artificial Intelligence and Statistics, 25-27 April 2023, Palau de Congressos, Valencia, Spain*, volume 206 of *Proceedings of Machine Learning Research*, pages 8586–8605. PMLR, 2023.

Sirui Chen, Bo Peng, Meiqi Chen, Ruiqi Wang, Mengying Xu, Xingyu Zeng, Rui Zhao, Shengjie Zhao, Yu Qiao, and Chaochao Lu. Causal evaluation of language models. *CoRR*, abs/2405.00622, 2024.

Zhuo Chen, Xinzhe Yuan, Jianshu Zhang, Jinzong Dong, Ruichen Zhou, Yingchun Niu, Tianhang Zhou, Yu Yang Fredrik Liu, Yuqiang Li, Nanyang Ye, and Qinying Gu. LABO: LLM-accelerated Bayesian optimization through broad exploration and selective experimentation. In *Forty-third International Conference on Machine Learning, ICML 2026, Seoul, South Korea, July 6-11, 2026*, volume 306 of *Proceedings of Machine Learning Research*, pages 17979–18005. PMLR, 2026.

Abdoulatif Cissé, Xenophon Evangelopoulos, Vladimir V. Gusev, and Andrew I. Cooper. Language-based Bayesian optimization research assistant (BORA). In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*, pages 4967–4975. ijcai.org, 2025.

Jean Durand, Yashas Annadani, Stefan Bauer, and Sonali Parbhoo. Causal Bayesian optimization with unknown graphs. *CoRR*, abs/2503.19554, 2025.

Muhammad Qasim Elahi, Mahsa Ghasemi, and Murat Kocaoglu. Partial structure discovery is sufficient for no-regret learning in causal bandits. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

Peter I. Frazier. A tutorial on Bayesian optimization. *CoRR*, abs/1807.02811, 2018.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona T. Diab, and Bernhard Schölkopf. Can large language models infer causation from correlation? In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *J. Glob. Optim.*, 13(4):455–492, 1998.

Emre Kiciman, Robert Osazuwa Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality. *Trans. Mach. Learn. Res.*, 2024, 2024.

Sanghack Lee and Elias Bareinboim. Structural causal bandits: Where to intervene? In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2573–2583, 2018.

Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance Bayesian optimization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Stephanie Long, Alexandre Piché, Valentina Zantedeschi, Tibor Schuster, and Alexandre Drouin. Causal discovery with language models as imperfect experts. *CoRR*, abs/2307.02390, 2023.

Jonas Mockus. On Bayesian methods for seeking the extremum. In *Optimization Techniques, IFIP Technical Conference, Novosibirsk, USSR, July 1-7, 1974*, volume 27 of *Lecture Notes in Computer Science*, pages 400–404. Springer, 1974.

Sumantrak Mukherjee, Mengyan Zhang, Seth R. Flaxman, and Sebastian J. Vollmer. Graph agnostic causal Bayesian optimisation. *CoRR*, abs/2411.03028, 2024.

Min Woo Park, Andy Arditi, Elias Bareinboim, and Sanghack Lee. Structural causal bandits under Markov equivalence. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.

Yusuf H. Roohani, Andrew H. Lee, Qian Huang, Jian Vora, Zachary Steinhart, Kexin Huang, Alexander Marson, Percy Liang, and Jure Leskovec. BioDiscoveryAgent: An AI agent for designing genetic perturbation experiments. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proc. IEEE*, 104(1):148–175, 2016.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press, 2000.

Scott Sussex, Anastasia Makarova, and Andreas Krause. Model-based causal Bayesian optimization. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

Scott Sussex, Pier Giuseppe Sessa, Anastasia Makarova, and Andreas Krause. Adversarial causal Bayesian optimization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Xinzhe Yuan, Zhuo Chen, Jianshu Zhang, Huan Xiong, Nanyang Ye, Yuqiang Li, and Qinying Gu. Unleashing LLMs in Bayesian optimization: Preference-guided framework for scientific discovery. In *The Fourteenth International Conference on Learning Representations, ICLR 2026, Rio de Janeiro, Brazil, April 23-27, 2026*. proceedings.iclr.cc, 2026.

A Analysis and Discussion

AGENTCBO targets graph beliefs that improve intervention selection through the ES–prior interface. The main text gives fixed-update guarantees and empirical analysis; this appendix records their scope and broader-impact considerations.

A.1 Limitations

We treat the following as open problems that bound the scope of our claims:

1. **Fixed-update scope.** At matched non-routing state and under common surrogate and selection rules, Lemma 1 and Proposition 1 establish fixed-update selector sufficiency and CEI stability. Extending these guarantees to the full adaptive run requires joint analysis of graph updates, archived ES data, and CEI allocation.
2. **Prior knowledge and evaluation scope.** Main-suite experiments use one LLM family, with a four-benchmark size comparison (Appendix E.5). Name obfuscation does not exclude training exposure. ALARM uses a public topology and real labels; ER supplies the generation ordering. These studies evaluate semantic-prior utility and operation under supplied structural information, not general causal reasoning with unknown orientations. The corruption study evaluates the gate's response to one prescribed erroneous update from a valid incumbent.
3. **Conditional refinement.** Calib and Full improve selected tasks, notably COMPLETEGRAPH. Suite-level tests do not establish consistent gains over Prior (Appendix C.2). Selecting refinement effort adaptively and measuring its marginal stacked cost remain useful extensions.

A.2 Broader Impacts

The main positive impact of AgentCBO is procedural: if graph discovery inside the loop can reduce the number of costly wet-lab or simulator interventions needed to reach a good operating point, then practitioners can save both time and experimental budget. The main negative risk is over-trust. In high-stakes domains such as medicine, policy, or biology, an incorrect discovered graph can still steer the optimization toward the wrong intervention family even when the final BO layer is well calibrated. We therefore view AgentCBO as a decision-support tool rather than an autonomous decision-maker, and we recommend human oversight plus benchmark-appropriate validation before any real deployment.

B Benchmark Suite

This appendix specifies the ten benchmarks used in the main experiments. Table 4 summarizes the key specifications; the flagship COMPLETEGRAPH benchmark is detailed in full below; the remaining nine benchmarks reuse the structural-equation models (SEMs) of their original sources, cited per row.

B.1 COMPLETEGRAPH: Full Specification, Flagship Benchmark

COMPLETEGRAPH embeds two latent variables U_1, U_2 that contaminate observational regressions involving the observed parents of Y. We use it as a hidden-confounder and finite-budget graph-misspecification stress test, examining the budget-allocation trade-off induced by changes in the runtime exploration family.

Structural equation model. The seven observed variables are $\{F, A, B, C, D, E, Y\}$ with manipulative subset $\mathbf{M} = \{B, D, E\}$ and target Y. The two latent variables U_1, U_2 and all structural noises

Table 4: Benchmark specifications. $|\mathbf{V}|$ is the number of observed variables; $|\mathbf{M}|$ is the number of manipulative variables; "ES" denotes the cardinality of the graph-derived RuntimeES under the known graph. "Direction" is the optimization direction; all ten benchmarks in this paper are run with `task=min`. The benchmark implementations are included in the supplementary material; this appendix summarizes the metadata needed to interpret the reported experiments and gives the full COMPLETEGRAPH SEM used for the hidden-confounder stress test.

Benchmark	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{M}\rvert$	ES	Direction	Manipulative variables	Source
CHAIN-5	5	1	1	min	X_0	`synthetic.py:chain`
TOYGRAPH	4	2	3	min	X, Z	`toy_graph.py`
PSAGRAPH	6	2	3	min	A, S	`psa_graph.py`
HIVEPIDEMIOLOGY	5	1	1	min	T	`hiv_epidemiology.py`
CCBOHEALTH	7	3	7	min	Aspirin, Statin, Cancer-Inhibitor	`ccbo_health.py`
COMPLETEGRAPH	7	3	7	min	B, D, E; latent U_1, U_2	`complete_graph.py`
PREDATORPREY	5	2	3	min	M, N	`predator_prey.py`
PISHCATGRAPH	11	2	3	min	P, I	`pishcat_graph.py`
ROSENBROCK	3	2	3	min	X_1, X_2	`acbo_synthetic.py:Rosenbrock`
ACKLEY	3	2	3	min	X_1, X_2	`acbo_synthetic.py:Ackley`

are independent standard normals.

$$
\begin{aligned}
F &\sim \mathcal{N}(0,1)\\
A &= F^2 + U_1 + \varepsilon_A\\
B &= U_2 + \varepsilon_B\\
C &= \exp(-B) + \varepsilon_C\\
D &= \exp(-C)/10 + \varepsilon_D\\
E &= \cos(A) + C/10 + \varepsilon_E\\
Y &= \cos(D) - D/5 + \sin(E) - E/4 + U_1 + \exp(-U_2) + \varepsilon_Y,
\end{aligned}
$$

with all $\varepsilon \sim \mathcal{N}(0,1)$.

Interpretation. Hard interventions on B, D, or E cut the incoming arrows into the intervened variables. The COMPLETEGRAPH experiments probe a finite-budget trade-off: omitting E from the discovered manipulative ancestry can concentrate the 50-step budget on a smaller family, while restricting useful joint-intervention opportunities. The resulting benefit depends on the retained sets and optimization seed.

Intervention ranges. $B \in [-5, 4]$, $D \in [-5, 5]$, and $E \in [-6, 3]$. The graph-derived runtime ES family contains all non-empty subsets of $\{B, D, E\}$:

$$\{\{B\}, \{D\}, \{E\}, \{B, D\}, \{B, E\}, \{D, E\}, \{B, D, E\}\}.$$

The discovered ESs $\widehat{\mathcal{E}}$ vary by run. When a committed graph keeps two manipulative ancestors, the constructor yields their three non-empty subsets and retains the omitted variable's singleton intervention, giving four sets instead of the seven known-graph runtime sets. The following controls separate family cardinality from composition and then trace a learned route under the known graph.

Cardinality-matched family controls. Exploration-set composition affects finite-budget outcomes even at fixed family size. We hold the COMPLETEGRAPH graph at ground truth and vary only the ES-family assignment rule, using exactly three sets per run, 20 matched seeds, $T = 50$, and $n_{\text{obs}} = 100$. The initialization, seed set, and remaining configuration are shared across arms. These three-set controls are distinct from the four-set family produced by the runtime constructor above.

Table 5 separates composition from cardinality: the fixed BD family reaches the low-objective regime on 14 seeds, whereas both E-containing families have zero hits throughout the threshold range -2.0 to -3.5. The random arm selects the BD family on eight seeds and reproduces the corresponding fixed-BD endpoints on those seeds; all four random-arm hits occur within this subset. One further non-BD draw ties the fixed-BD endpoint. For fixed BD versus fixed BE and versus the

Table 5: Fixed-graph, cardinality-matched CompleteGraph controls. Every arm receives three exploration sets and the same budget over 20 matched seeds. A low-objective hit is a best observed $y \leq -3.0$.

ES-family assignment rule	Low-objective hits
Fixed $\{B\}, \{D\}, \{B, D\}$	$14/20$ (70%)
Random pair-induced family per seed	$4/20$ (20%)
Fixed $\{B\}, \{E\}, \{B, E\}$	$0/20$
Fixed $\{D\}, \{E\}, \{D, E\}$	$0/20$

Table 6: Endpoints for one successful CompleteGraph run, its matched-seed references, and route-only known-graph replays. The replay rows change only the RuntimeES schedule while recomputing downstream optimization. All three contracted schedules improve on the native known-graph reference.

Experiment	Configuration	Endpoint
Traced run	AgentCBO	-3.51
Matched-seed reference	AgentCBO-Prior	-2.20
Matched-seed reference	Native known-graph CBO	-2.07
Known-graph replay	Learned RuntimeES schedule (BD)	-3.28
Known-graph replay	Matched switch/cardinality (BE)	-2.95
Known-graph replay	Matched switch/cardinality (DE)	-2.34

random arm, exact two-sided McNemar tests on the hit indicator give $p = 1.2 \times 10^{-4}$ and 0.0020, respectively. Exact two-sided paired Wilcoxon tests on the continuous endpoint give $p = 1.5 \times 10^{-5}$ and 9.8×10^{-4}, omitting zero differences and leaving effective sample sizes 18 and 11.

Realized route and known-graph replay. On one successful traced run, a calibrated graph passes the commit gate at outer-loop step 15, contracting RuntimeES from seven sets to $\{B\}, \{D\}, \{E\}, \{B, D\}$. Among actual intervention decisions, selection of $\{B, D\}$ rises from $1/11$ (9%) before commitment to $18/27$ (67%) afterward. The other 12 of the 50 outer-loop steps are observations. The first outcome $y \leq -3.0$ occurs at step 29, and the run ends at -3.51, versus -2.20 for matched-seed Prior and -2.07 for native known-graph CBO.

The route-only replay retains the known graph and changes only the RuntimeES schedule. Seed, initialization, budget, observation/intervention timing, and true-graph do-prior construction are fixed; acquisition scores, selected ESs, intervention values, outcomes, and surrogate updates are recomputed. The fitted predictive trajectory evolves with this recomputed downstream optimization. Alternative schedules retain $\{B, E\}$ or $\{D, E\}$ instead of $\{B, D\}$, with the same switch time and cardinality.

The learned-route replay recovers approximately 84% of this seed's AgentCBO–known-graph endpoint gap (Table 6). All three contractions improve on native known-graph CBO, with BD strongest. Together with the cardinality-matched controls, this trace connects the finite-budget improvement to both family contraction and retained-pair composition.

B.2 Other Benchmarks

The remaining nine benchmarks reuse SEMs from their original sources; we list brief structural descriptions and cite the source families here. Chain-5, ToyGraph, and PSAGraph follow the original CBO suite [Aglietti et al., 2020]; CCBOHealth, PredatorPrey, and PISHCATGraph follow the constrained-CBO, dynamic-CBO, and MCBO benchmark suites, respectively [Aglietti et al., 2023, 2021, Sussex et al., 2023].

- Chain-5: a 5-node chain $X_0 \to X_1 \to X_2 \to X_3 \to Y$ with sinusoidal/exponential noise and a single manipulative variable $X_0 \in [-3, 3]$.
- ToyGraph: 4 observed variables $\{C, X, Y, Z\}$ with manipulative $\{X, Z\}$; two-edge structure $X \to Y$ and $Z \to Y$ with confounder C.

- PSAGRAPH: PSA-medication graph with two manipulative variables, Aspirin dose A and Statin dose S, and target PSA-level reduction.
- HIVEPIDEMIOLOGY: 5-node HIV treatment benchmark with single manipulative variable T, treatment intensity in $[0, 1]$, and target $Y = -\big(\cos(3L) + \sin(3R) + L + R - 0.5B + \varepsilon_Y\big)$ where L is viral load, R is drug resistance, and B is behavioural risk. SEM in `hiv_epidemiology.py`.
- CCBOHEALTH: cardiovascular-risk benchmark with manipulative variables Aspirin dose, Statin dose, and Cancer-Inhibitor; 7 observed variables.
- PREDATORPREY: 5-node ecological dynamics benchmark with manipulative variables M, mortality rate, and N, carrying-capacity input; adapted to the static setting.
- PISHCATGRAPH: 11-node multi-ancestor benchmark with manipulative variables P and I and runtime family $\{\{P\}, \{I\}, \{P, I\}\}$; designed to test multi-ancestor performance.
- ROSENBROCK, ACKLEY: 2D optimization surfaces adapted from the ACBO synthetic benchmark suite as causal minimization problems. ROSENBROCK uses a sign-inverted, offset variant of the standard Rosenbrock function with a non-trivial linear causal link $X_1 \to X_2$ in addition to $X_1, X_2 \to Y$:

$$X_1 = \omega_1, \quad X_2 = 0.6\, X_1 + 0.4\, \omega_2,$$
$$Y = -100\,(X_2 - X_1^2)^2 - (1 - X_1)^2 + 10 + \omega_Y.$$

  ACKLEY uses a simplified, separable variant with the trivial DAG $X_1, X_2 \to Y$ and no $X_1 \to X_2$ link:

$$Y = 20\, \exp\Big(-0.2\, \sqrt{|X_1|}\Big) + \exp(X_2) - 20 - e + \omega_Y.$$

  In both cases the optimizer minimizes Y over X_1, X_2 on the intervention ranges $X_1, X_2 \in [0, 1]$ (ROSENBROCK) and $X_1, X_2 \in [-2, 2]$ (ACKLEY). Implementation: `acbo_synthetic.py`.

The benchmark implementations corresponding to this suite are included in the supplementary material; this appendix summarizes the benchmark metadata used to interpret the reported experiments and provides the full COMPLETEGRAPH SEM that underlies the main hidden-confounder analysis.

C Additional Results and Diagnostics

C.1 Matched Dynamic PC-Only Ablation

The PC-then-CBO baseline, executable alias `UnknownDAG-CBO`, is a matched dynamic PC-only ablation of AGENTCBO-PRIOR. Both instantiate the same AGENTCBO outer loop; only `LPEngine` is replaced by `PCOnlyEngine`. They share the optimization budget, update frequency, initialization coverage fallback, periodic commit gate, RuntimeES/do-prior rebuilding, archive reuse, and acquisition loop. We use the name *dynamic PC-only* to distinguish this periodic guarded baseline from one-shot PC preprocessing. Table 7 reports the full comparison.

C.2 Suite-Level Paired Comparisons

The suite-level comparison pairs the ten benchmark means, rather than pooling individual optimization runs. For each comparator, let Δ be the endpoint mean of AGENTCBO-PRIOR minus that of the comparator. All tasks minimize the objective, so negative differences favor Prior and W^+ sums positive ranks. We use exact two-sided sign permutations of the nonzero paired differences for the Wilcoxon signed-rank statistic, assigning average ranks to tied absolute differences. Effective sample sizes range from eight to ten. Holm correction uses the unrounded nominal p-values of all nine comparisons: seven reference methods and the two additional refinement variants.

Prior has better endpoint means on eight to ten benchmarks against each reference method (Table 8); the comparisons with BO and GACBO remain significant after Holm correction. The stage-wise tests detect no suite-level difference between Prior and Calib ($p = 0.84$), or Prior and Full ($p = 0.83$). These paired tests assess suite-level differences across benchmark means.

Table 7: Matched dynamic PC-only ablation (previously PC-then-CBO), executable alias `UnknownDAG-CBO`, versus AGENTCBO-PRIOR on the full ten-benchmark suite. All benchmarks are minimization, so $\Delta :=$ PC-only mean $-$ LP mean: positive values favor LP. The strongest LP gains occur on ROSENBROCK, COMPLETEGRAPH, PSAGRAPH, and PREDATORPREY; AGENTCBO-PRIOR ties with PC-only on CHAIN-5. The only counterexample is PISHCATGRAPH, where PC-only marginally outperforms AGENTCBO-PRIOR ($\Delta = -0.02$).

Benchmark	AGENTCBO-PRIOR	Dynamic PC-only	Δ, favors LP
COMPLETEGRAPH	-2.50 ± 1.21	-2.13 ± 1.45	$+0.38$
TOYGRAPH	-2.11 ± 0.17	-2.02 ± 0.16	$+0.09$
PSAGRAPH	$+5.18 \pm 0.07$	$+5.35 \pm 0.13$	$+0.17$
CCBOHEALTH	$+4.96 \pm 0.09$	$+4.98 \pm 0.09$	$+0.02$
HIVEPIDEMIOLOGY	-1.50 ± 0.01	-1.48 ± 0.01	$+0.02$
CHAIN-5	-1.32 ± 0.01	-1.32 ± 0.01	$+0.00$
ROSENBROCK	-90.98 ± 0.05	-89.80 ± 0.02	$+1.18$
ACKLEY	-7.51 ± 0.01	-7.49 ± 0.01	$+0.02$
PREDATORPREY	-2.31 ± 0.01	-2.17 ± 0.22	$+0.14$
PISHCATGRAPH	$+0.30 \pm 0.10$	$+0.28 \pm 0.01$	-0.02

Table 8: Paired suite-level tests for AGENTCBO-PRIOR, with benchmark means as the pairing unit. The seven reference-method rows are shown; Holm adjustment includes all nine comparisons, including Calib and Full. Wins, ties, and losses are counted from Prior's perspective.

Comparator	W^+	Nominal p	Wins / ties / losses	Holm p
BO	0	0.0020	10 / 0 / 0	0.018
GACBO	0	0.0020	10 / 0 / 0	0.018
CEO	0	0.0078	8 / 2 / 0	0.055
LLAMBO	3	0.0098	9 / 0 / 1	0.059
CBO-U	5	0.0195	8 / 0 / 2	0.098
BORA	4	0.0273	8 / 1 / 1	0.109
Known-graph CBO	5	0.0352	8 / 1 / 1	0.109

C.3 Routing Diagnostics

RuntimeES and the fitted prior mean. Across 376 systematically edited graph pairs from six benchmarks, 323 pairs preserve RuntimeES. Among those 323 pairs, 141 (44%) exceed the fitted prior-mean-distance threshold $\delta_\mu = 0.05$, with a maximum reported distance of 2.03 objective units. As δ_μ increases from 0.01 to 0.5, the fraction above threshold decreases from 69% to 26%. Ancestor-subgraph nSHD has a moderate association with fitted prior-mean distance (Spearman $\rho = 0.51$). These diagnostics show why RuntimeES equality alone does not determine the prior-mean channel. They measure fitted means, whereas the exact fixed-update interface also includes graph-dependent uncertainty inputs (Appendix G).

Single-edge perturbations. A separate study deletes one edge on a directed path from a manipulative variable to Y in each of the ten benchmarks. Each perturbed arm is compared against its own matched unperturbed ground-truth-graph control, using 20 matched seeds per cell; these controls are separate from the main result matrix. Seven perturbations change RuntimeES. Five of them produce endpoint degradations that remain significant after Bonferroni correction across all ten perturbations, with adjusted $p \leq 0.0065$ (Table 9). The three RuntimeES-preserving perturbations produce only small endpoint movements in this study. An edge deletion can change both RuntimeES and do-priors, so these contrasts measure end-to-end routing sensitivity rather than an isolated ES effect.

C.4 Controlled Update-Time Gate Stress Test

The PSAGRAPH stress test measures whether an erroneous LP update can overwrite a valid committed route. We elicit one new pairwise LP matrix with the same prompt and model as in the benchmark experiments, at temperature zero, and freeze it. A prespecified, endpoint-blind structural rule selects

Table 9: The five significant single-edge perturbations from the ten-benchmark study, each with 20 matched seeds. Displayed p-values are the reported exact two-sided paired Wilcoxon values before Bonferroni adjustment; all five remain significant after correction across ten comparisons. Lower endpoints are better.

Benchmark	Unperturbed $\rightarrow$ perturbed	Paired p	Seeds worse
ROSENBROCK	$-90.97 \rightarrow -44.78$	1.9×10^{-6}	$20/20$
ACKLEY	$-7.50 \rightarrow -6.05$	1.9×10^{-6}	$20/20$
PISHCATGRAPH	$+0.21 \rightarrow +0.39$	1.9×10^{-6}	$20/20$
CCBOHEALTH	$+5.11 \rightarrow +5.45$	1.05×10^{-4}	$17/20$
COMPLETEGRAPH	$-2.497 \rightarrow -1.334$	6.4×10^{-4}	$15/20$

Table 10: PSAGRAPH gate stress test from a common valid incumbent, with coverage rescue inactive and singleton completion active. Endpoint differences and 95% confidence intervals use the ten matched seed differences. A positive difference indicates degradation for corrupted-gated versus clean-gated, and favors the gate for corrupted-bypassed versus corrupted-gated.

Paired contrast	Endpoint Δ	95% CI	$+/-/\text{tie}$
Corrupted-gated $-$ clean-gated	0	$[0, 0]$	$0/0/10$
Corrupted-bypassed $-$ corrupted-gated	$+0.1387$	$[+0.0086, +0.2839]$	$10/0/0$

the first corruption that, after cycle resolution, changes exactly four high-confidence pair states and strictly increases pair-state error against the true graph. The four changes comprise two direction reversals, one deletion, and one spurious addition.

The clean-gated, corrupted-gated, and corrupted-bypassed arms start from the same valid incumbent, committed once from the clean matrix and verified to cover both manipulative ancestors of the target. They share matched observations, seeded exogenous streams, budget, and implementation over ten held-out matched seeds, giving 30 evaluation runs. The bypass arm commits candidates marked as changed by the shared detector while skipping the confidence, manipulative-ancestor, and nSHD checks. Coverage rescue (the coverage fallback) is inactive in all three arms, while mandatory singleton completion remains active. This isolates update-time commitment from initialization failure and rescue behavior.

With the full gate active, corruption changes the candidate proposal in all ten seeds while preserving committed RuntimeES and the fitted-prior diagnostic, with zero endpoint degradation (Table 10). Holding the corrupted proposal fixed, bypassing the gate first changes the commit decision, then changes both measured routing diagnostics and worsens all ten endpoints. In this prescribed update-time stress test, the gate preserves both measured routing diagnostics and prevents endpoint degradation.

C.5 Larger-Graph Studies

The ALARM study isolates semantic-prior utility on a published 37-node topology; the ordered Erdős–Rényi (ER) study measures optimization and computational cost through 50 nodes under a supplied ordering convention.

C.5.1 ALARM: Semantic-Prior Isolation

We use the published ALARM topology distributed by bnlearn, with 37 nodes, 46 directed arcs, and the original variable labels [Beinlich et al., 1989]. The synthetic linear SEM has seed-fixed random-sign coefficients with magnitudes drawn from $U(0.5, 1.5)$ and noise standard deviation 0.3. The manipulators are `FIO2`, `MVS`, `VMCH`, `ANES`, and `INT`, each with domain $[-3, 3]$; the target is `BP`. Their 31-set intervention family has interaction headroom of 3.36 target units over the best singleton. One elicited prior is frozen throughout the study and recovers 43 of the 46 true arcs with direction.

With coverage rescue inactive and mandatory singleton completion retained, we compare the prior and dynamic PC-only arms over ten matched seeds, with a budget of 50 trials per run. The prior

Table 11: ALARM component isolation over ten matched seeds. Both arms retain mandatory singleton completion. Endpoints use the known-family-optimum/best-singleton normalization described in the text; lower is better. PC-only's reported spread is 0.001. The default rescue-active setting is summarized by the paired tie.

Coverage rescue	Prior	PC-only	Paired comparison
Inactive	31 sets mean 0.23	5 sets score 1.001	10/10 prior wins $p = 0.002$
Active (default)	Tie		No measurable separation

retains all 31 sets through one accepted commit. PC discovery produces no manipulative ancestor of the target, so the gate rejects its candidates and singleton completion supplies five sets. Table 11 normalizes endpoints so that 0 is the known-family optimum and 1 is the best-singleton value. The prior's mean is 0.23, compared with PC-only's reported score of 1.001, and the prior improves all ten paired endpoints (two-sided sign test $p = 0.002$). Enabling the default coverage rescue makes the arms tie. Thus, the isolation identifies the prior's contribution to multi-variable routing, while the default rescue protects against the PC discovery failure.

C.5.2 Ordered ER Graphs: Protocol, Outcomes, and Cost

Graph generation and matched protocol. For each $n \in \{10, 20, 50\}$, we generate ten independent ordered ER DAGs and use two matched optimization seeds per DAG. Edges point from lower to higher index with probability $2/n$, with a target parent added when necessary. The linear additive-noise SEM uses signed weights with magnitudes drawn from $U[0.5, 2]$, root/non-root noise standard deviations of $1/0.1$, and the final node as target Y. Every non-target variable is intervenable over $[-3, 3]$. $\mathrm{RuntimeES}$ selects the largest interaction order whose cumulative family fits a 250-set budget, then includes every manipulator singleton. The 250-set budget therefore precedes mandatory singleton completion.

All arms use $T = 50$ and $n_{\mathrm{obs}} = 100$: AGENTCBO with a frozen generic indexed-label prior (Agent), dynamic PC-only AGENTCBO (PC), and known-graph CBO (Known), a finite-budget reference using the same family-construction rule. The LLM receives indexed labels, the target label, and the lower-to-higher ordering convention, but no realized adjacency, weights, samples, or outcomes. We elicit one prior per size at temperature zero and reuse it across that size's DAGs and optimization seeds. The study therefore evaluates use of a disclosed structural prior at scale.

Metric and aggregation. Each returned intervention $\hat{a}$ is evaluated on the noise-free SEM for DAG g. The normalized known-family gap is

$$\mathrm{NG}_g(\hat{a}) = \frac{\mu_g(\hat{a}) - y^\star_{K,g}}{\mu_g(\emptyset) - y^\star_{K,g}},$$

where $\mu_g(\emptyset)$ is the no-intervention target value and $y^\star_{K,g}$ is the analytic optimum over the known-graph $\mathrm{RuntimeES}$. Lower is better. This denominator differs from the best-singleton reference used for ALARM. We average the two optimization seeds within each DAG, then report each method's DAG-level median and interquartile range (IQR). The paired effect Δ is the mean of DAG-level Agent-minus-PC differences, not the difference between the displayed medians; 95% confidence intervals resample DAGs.

Every arm completed both seeds on all ten DAGs at every size. All prespecified DAGs are retained, including those without interaction headroom. The numbers with $y^\star_{\mathrm{singleton},g} - y^\star_{K,g} > 10^{-10}$ are 6, 9, and 8 at 10, 20, and 50 nodes, respectively. Table 12 reports the complete comparison. Paired point estimates favor Agent at 20 and 50 nodes, while every single-size interval includes or touches zero at the reported precision. At 20 nodes, Agent wins on three DAGs and ties on seven; at 50 nodes it wins on five, loses on one, and ties on four.

Computational cost. Table 13 reports per-run optimization wall-clock time and the largest visited family; Table 14 separates the one-time prior elicitation. The two discovery arms trigger coverage

Table 12: Ordered ER outcomes over ten DAGs per size, with two optimization seeds averaged within each DAG. Method columns show NG median [IQR]; Δ is the mean paired Agent-minus-PC difference, with a DAG-resampled 95% CI. Negative Δ favors Agent. Known denotes the finite-budget known-graph reference.

Nodes	Agent NG	PC NG	Known NG	Δ	95% CI	Agent wins /10
10	0.0415 [0.0001, 0.1393]	0.0417 [0.0000, 0.1393]	0.0000 [0.0000, 0.0008]	$+0.0219$	$[-0.0001, +0.0658]$	2
20	0.3575 [0.2496, 0.4756]	0.4143 [0.2568, 0.4771]	0.0002 [0.0000, 0.1536]	-0.0142	$[-0.0420, +0.0000]$	3
50	0.4568 [0.3623, 0.5715]	0.6312 [0.5440, 0.7507]	0.0668 [0.0127, 0.2236]	-0.1287	$[-0.1932, +0.0022]$	5

Table 13: Ordered ER optimization cost over twenty runs per arm and size. Wall-clock hours are per-run median [IQR], excluding one-time prior elicitation. Largest family is the cross-run median [maximum] of each run's largest visited family. Mandatory singleton completion follows the 250-set interaction-family budget.

	Optimization time (hours)			Largest family		
Nodes	Agent	PC	Known	Agent	PC	Known
10	1.57 [0.67, 2.28]	0.95 [0.49, 2.87]	0.75 [0.53, 1.00]	10 [129]	15 [129]	16.5 [66]
20	3.79 [3.16, 5.48]	5.83 [4.85, 9.47]	6.13 [5.17, 8.61]	110 [190]	110 [190]	60.5 [257]
50	8.17 [6.28, 11.31]	9.27 [8.39, 11.57]	6.19 [5.05, 7.88]	49 [185]	49 [50]	75 [169]

fallback equally often: 9/20, 10/20, and 10/20 runs at 10, 20, and 50 nodes. Agent's median optimization time is about 35% lower than PC-only at 20 nodes and 12% lower at 50 nodes. Amortizing the 50-node elicitation over its twenty runs reduces the latter saving to about 10%. Together, these results establish operation and quantify cost through 50 nodes under the supplied ordering prior.

D Shared Runtime and Secondary Modules

D.1 Runtime Routing Definitions and Full Gate Predicate

Section 4.2 uses compact runtime definitions. Here we record the corresponding exact objects used for reproducibility. Let $\mathcal{I}$ denote the manipulative set, equivalently M in the benchmark descriptions, and write $\mathcal{A}_Y(\mathcal{G}) = \mathrm{anc}_{\mathcal{G}}(Y) \cap \mathcal{I}$. The ideal CBO family is $\mathcal{ES}_{\mathrm{ideal}}(\mathcal{G}) = 2^{\mathcal{A}_Y(\mathcal{G})} \setminus \{\emptyset\}$. The standard runtime constructor is $\widehat{\mathcal{E}}(\mathcal{G}) = \mathrm{RuntimeES}(\mathcal{G}; \mathcal{I}, Y)$: it keeps singleton manipulative interventions, truncates very large families, and falls back to a conservative default when no manipulative ancestor is discovered. The component-isolation studies explicitly disable coverage rescue while retaining singleton completion (Appendices C.4 and C.5.1).

For a local edit Δ, the do-prior distance used to instantiate Definition 2 is

$$d_\mu(\mathcal{G}, \mathcal{G} \oplus \Delta) := \max_{S \in \widehat{\mathcal{E}}(\mathcal{G}) \cap \widehat{\mathcal{E}}(\mathcal{G} \oplus \Delta)} \sup_{\mathbf{x} \in \mathcal{D}(S)} \left| \mu_S^{\mathrm{prior}}(\mathcal{G})(\mathbf{x}) - \mu_S^{\mathrm{prior}}(\mathcal{G} \oplus \Delta)(\mathbf{x}) \right|.$$

The main text uses this quantity only as a conceptual routing-criticality criterion; the implemented gate uses the cheaper proxy below.

Let $\tilde{\mathcal{G}}_t$ be the candidate graph returned by the active stack, c_t its confidence, and $A_t := \mathrm{anc}_{\hat{\mathcal{G}}_{t-1}}(Y) \cup \mathrm{anc}_{\tilde{\mathcal{G}}_t}(Y) \cup \{Y\}$. Let $n\mathrm{SHD}_t := \mathrm{SHD}(\hat{\mathcal{G}}_{t-1}|_{A_t}, \tilde{\mathcal{G}}_t|_{A_t}) / \binom{|A_t|}{2}$ denote the SHD between the two ancestor-subgraphs on A_t, normalized by the maximum possible undirected edges. With $\tau_c = 0.5$ and $\Phi_t := \min(\tau_c + \frac{1}{2} n\mathrm{SHD}_t, 0.9)$, the full acceptance predicate is

$$c_t \geq \tau_c \quad \wedge \quad \tilde{\mathcal{G}}_t \neq \hat{\mathcal{G}}_{t-1} \quad \wedge \quad \mathcal{A}_Y(\tilde{\mathcal{G}}_t) \neq \emptyset \quad \wedge \quad c_t \geq \Phi_t. \tag{10}$$

Table 14: One-time prior elicitation for the ordered ER study. Each size-specific prior is shared by all ten DAGs and twenty optimization runs. These costs are separate from Table 13.

Nodes	Prior time (seconds)	Prior tokens
10	304	13,130
20	1,262	55,744
50	12,190	360,360

D.2 Language-Prior Agent Executable Details

Algorithm 2 specifies the LP executable engine that implements the LANGUAGE-PRIOR AGENT referenced in Section 4.3.

Algorithm 2 LANGUAGE-PRIOR AGENT (LP executable): LLM–PC graph proposal

Require: $\mathcal{D}_{\text{obs}}$: observational data; $\mathbf{V}$: variable set; $\kappa \in \mathcal{K}$: domain context
Require: $\alpha = 0.05$, PC significance; $\tau_{\text{req}} = 0.7$; $\tau_{\text{sup}} = 0.8$; $n_{\text{PC}} = 30$, sample-complexity floor
Require: $\tau_{\text{forbid}} = 0.2$, forbidden-edge threshold
Ensure: proposed DAG $\hat{\mathcal{G}}$; frozen backbone $\mathcal{B}$; residual pairs $\mathcal{R}$; confidence $c \in [0.1, 0.95]$

Stage I. LLM prior elicitation

1: Query the LLM for each ordered pair (i, j) to obtain pairwise probability matrix $P \in [0,1]^{|\mathbf{V}|\times|\mathbf{V}|}$
2: Resolve cycles in P by greedily weakening the weakest edge in each cycle to $p = 0.4$
3: Build background knowledge: required edges $\mathcal{B}^+ = \{(i,j) : P_{ij} > \tau_{\text{req}} \wedge P_{ji} < 1 - \tau_{\text{req}}\}$ and forbidden edges $\mathcal{B}^- = \{(i,j) : P_{ij} < \tau_{\text{forbid}} \wedge P_{ji} < \tau_{\text{forbid}}\}$

Stage II. Constraint-based anchoring

4: Run PC on $\mathcal{D}_{\text{obs}}$ with background knowledge $(\mathcal{B}^+, \mathcal{B}^-)$ at significance α, yielding a completed partially directed acyclic graph (CPDAG) $\mathcal{G}_{\text{PC}}$
5: Extend $\mathcal{G}_{\text{PC}}$ to a DAG $\hat{\mathcal{G}}_{\text{PC}}$ via the standard `pdag2dag` extension routine, which orients remaining undirected edges in $\mathcal{G}_{\text{PC}}$ while preserving acyclicity and not introducing new v-structures; on timeout or extension failure, fall back to a deterministic low-index-to-high-index tie-break (no LLM involvement at this step)
6: For each (i, j) with $P_{ij} > \tau_{\text{sup}}$ that is absent from $\hat{\mathcal{G}}_{\text{PC}}$, add edge $i \to j$ if it introduces no cycle and the conditional-independence test on (i, j) given PC's separating set $\text{sepset}_{\text{PC}}(i, j)$ rejects independence at level 3α (i.e., $p_{i,j|\text{sepset}_{\text{PC}}(i,j)} \leq 3\alpha$)

Stage III. Confidence scoring and residual partition

7: $c \leftarrow \text{clip}\Big(\frac{1}{2} \cdot \frac{n_{\text{obs}}}{n_{\text{obs}}+n_{\text{PC}}} + \frac{1}{2} \cdot \text{agreement}(P, \hat{\mathcal{G}}),\ 0.1,\ 0.95\Big)$
8: Compute frozen backbone $\mathcal{B}$ and residual pairs $\mathcal{R}$ as defined in Eq. (4)
9: **return** $(\hat{\mathcal{G}},\ \mathcal{B},\ \mathcal{R},\ c)$

Agreement functional and threshold design. We define $\text{agreement}(P, \hat{\mathcal{G}}) := |\{(i,j) \in E(\hat{\mathcal{G}}) : P_{ij} > 0.5\}|/|E(\hat{\mathcal{G}})|$ as the fraction of discovered edges whose direction is endorsed by the LLM prior, with agreement $:= 0$ if $E(\hat{\mathcal{G}}) = \emptyset$. The required-edge threshold $\tau_{\text{req}} = 0.7$ demands two-sided LLM agreement before writing a pair into PC's background knowledge; the supplementation threshold $\tau_{\text{sup}} = 0.8$ admits LLM-supported pairs that PC omitted, subject to a relaxed CI guard at level 3α that prevents blind faith in the LLM. Forbidden edges fire only when both directions of a pair score below $\tau_{\text{forbid}} = 0.2$. The fully-connected fallback triggers only at initialization when the very first discovery returns $\mathcal{A}_Y(\hat{\mathcal{G}}) = \emptyset$; update-time candidates with the same defect are rejected by the outer gate. All thresholds were fixed a priori on a three-benchmark pilot. Appendix E reports sensitivity over α, T_{LLM}, and T_{update}.

D.3 Secondary Modules and Standalone Diagnostics

This subsection summarizes the secondary modules used in the stacked variants and standalone module diagnostics. The LP engine implements the LANGUAGE-PRIOR AGENT and remains the

proposal-only AgentCBO variant covered in Section 4.3; the goal here is to document the remaining modules well enough for reproduction and for the standalone diagnostics reported in Section 5.3.

Calibration Agent executable alias, CC. The CC executable engine implements the CALIBRATION AGENT role in the stacked variants. It is a seven-step pipeline that uses the LLM as an orchestrator rather than a direct graph editor. It first detects coarse dataset properties (linearity and Gaussianity), asks the LLM to select a small candidate pool from {PC, GES, DirectLiNGAM}, requests per-algorithm hyperparameters, executes the selected methods, reranks the resulting graphs using a weighted score over fit, structure, Bayesian information criterion (BIC), and stability, and finally applies bootstrap pruning plus DAG enforcement. Standalone CC is a diagnostic engine, not evidence for the stacked AGENTCBO-CALIB variant. The stacked AGENTCBO-CALIB variant uses the same outer-loop interface but treats CC-style orchestration as a calibration module on top of the LP graph.

Residual Repair Agent executable alias, RA. The RA executable engine implements the RESIDUAL REPAIR AGENT role in the stacked variants. It uses a PC-initialized graph and then gives the LLM access to a small set of causal-analysis tools, including variable summaries, pairwise correlations, CI tests, graph edits, collider/confounder checks, BIC, and acyclicity checks. The agent can refine the graph for up to a fixed number of ReAct iterations before returning a finalized adjacency matrix and a scalar confidence. Standalone RA is a diagnostic engine, not evidence for the stacked AGENTCBO-FULL variant. The stacked AGENTCBO-FULL variant uses RA-style tool use as a bounded repair module after LP initialization and CC-style calibration.

Implementation and compute environment. All LLM-assisted discovery runs use Qwen3.5-35B-A3B[1] served locally through an OpenAI-compatible endpoint backed by vLLM[2]. The OpenAI Python client is used to issue chat-completion requests; sampling temperature is fixed at $T_{\text{LLM}} = 0.5$ for reported runs unless stated otherwise (sensitivity in Appendix E). The frozen PSA and ER priors use temperature zero (Appendices C.4 and C.5.2). LP/CC/RA sweeps and CEO ran on A100/H200-class GPU workers; graph-free baselines used the same local environment when they required the LLM, while GACBO ran on high-memory CPU workers. Main-suite methods share a budget of 50 outer-loop steps after 100 observational samples per seed; the additional controls and larger-graph protocols are specified separately in Appendix C.

Module discovery cost profile. Table 15 reports per-run discovery cost for the three executable engines, averaged over all ten benchmarks.

Table 15: Module diagnostic cost table, averaged over all 10 benchmarks. AGENTCBO-PRIOR uses only the proposal layer. Standalone CC and RA are separate diagnostic engines.

	AGENTCBO-PRIOR	CC diagnostic	RA diagnostic
LLM calls per run	**12**	30	95
LLM tokens per run	**4.1 K**	7.3 K	139 K
Discovery time, seconds	**137**	197	1,138
Relative LLM cost	**1×**	1.8×	34×

E LP Sensitivity and Diagnostic Checks

This appendix documents the sensitivity of AGENTCBO-PRIOR to its discovery-engine hyperparameters and reports empirical diagnostics for gate activity.

E.1 Hyperparameter Sensitivity

The LP engine exposes three families of hyperparameters that could plausibly influence final $Y^\star$: the PC significance level α used in the constraint-based anchoring and edge-supplement steps, the LLM sampling temperature T_{LLM} used for pairwise prior elicitation, and the CBO outer-loop

[1] Open-weights model card: `https://huggingface.co/Qwen/Qwen3.5-35B-A3B`, Apache 2.0 license.
[2] Inference server: `https://github.com/vllm-project/vllm`, Apache 2.0 license.

rediscovery period T_{update} of Algorithm 1. We sweep α and T_{LLM} on two representative benchmarks spanning two empirical patterns in the suite: COMPLETEGRAPH, a dense hidden-confounder setting, and PSAGRAPH, an application-motivated multi-ancestor setting. Table 16 reports mean $\pm$ std of $Y^\star$ at $n = 20$ seeds per cell. The rediscovery period T_{update} is evaluated separately on the full ten-benchmark suite in §E.3.

Table 16: Sensitivity of AGENTCBO-PRIOR final $Y^\star$, reported as mean $\pm$ std over 20 seeds, to two engine hyperparameters. Default values used in the main experiments: $\alpha = 0.05$, $T_{\text{LLM}} = 0.5$. Confidence thresholds $\tau_{\text{req}} = 0.7$, $\tau_{\text{sup}} = 0.8$ are held fixed throughout this submission; the main claims do not rely on a separate full sweep over those two thresholds.

Setting	COMPLETEGRAPH, min	PSAGRAPH, min
PC $\alpha = 0.01$, stricter	-1.76 ± 0.92	$+5.33 \pm 0.15$
PC $\alpha = 0.05$, default	-2.50 ± 1.21	$+5.18 \pm 0.07$
PC $\alpha = 0.10$, looser	-1.81 ± 0.91	$+5.33 \pm 0.15$
LLM temperature $T_{\text{LLM}} = 0.3$	-2.35 ± 1.43	$+5.27 \pm 0.17$
LLM temperature $T_{\text{LLM}} = 0.5$, default	-2.50 ± 1.21	$+5.18 \pm 0.07$
LLM temperature $T_{\text{LLM}} = 0.7$	-1.65 ± 1.51	$+5.17 \pm 0.07$

Two observations follow. First, α is the less sensitive of the two parameters: stricter $\alpha = 0.01$ and looser $\alpha = 0.10$ choices change the COMPLETEGRAPH mean by ≤ 0.74 and leave PSAGRAPH essentially unchanged. Second, T_{LLM} matters only when statistical evidence is unreliable: PSAGRAPH fluctuates by at most 0.09 across all sampled temperatures (well within its standard deviation), while COMPLETEGRAPH (whose hidden confounders compromise PC's CI tests) shows the largest temperature sensitivity (at $T_{\text{LLM}} = 0.7$ the mean is -1.65 vs. a default of -2.50). High-temperature sampling produces contradictory pairwise probabilities that the cycle-resolution heuristic discards; this only matters when LP cannot fall back on reliable CI evidence.

E.2 Acceptance-Gate Frequency

We report how often the implemented outer gate accepts a rediscovered graph during a 50-step run. Table 17 reports, for each benchmark and each of 20 seeds, the fraction of rediscovery rounds at which the gate accepted the candidate graph.

Table 17: Acceptance-gate trigger rate, the fraction of rediscovery rounds at which a graph refinement is accepted, aggregated over 20 seeds per benchmark. With $T = 50$ and $T_{\text{update}} = 5$ there are at most 10 rediscovery rounds per run; the "accepted" count is the number at which the plausible-graph criterion of Eq. (10) holds.

Benchmark	Mean acceptance rate	Median # accepted rounds
COMPLETEGRAPH	$\sim 33\%$	3 of 10
TOYGRAPH	$\sim 22\%$	2 of 10
PSAGRAPH	$\sim 28\%$	2 of 10
CCBOHEALTH	$\sim 22\%$	2 of 10
HIVEPIDEMIOLOGY	$\sim 11\%$	1 of 10
CHAIN-5	$\sim 11\%$	1 of 10
ROSENBROCK	$\sim 11\%$	1 of 10
ACKLEY	$\sim 22\%$	2 of 10
PREDATORPREY	$\sim 32\%$	3 of 10
PISHCATGRAPH	$\sim 22\%$	2 of 10

The gate is selective: a median of three or fewer accepted refinements per run across all ten benchmarks, with rates bounded between 11% and 33%. Higher acceptance rates correlate with benchmarks where the initial observational sample produces a noisier skeleton: COMPLETEGRAPH has the highest rate (33%) because PC's CI tests struggle under hidden confounding, so new observational data in subsequent rounds continues to refine the skeleton, and PREDATORPREY (32%) shows similar refinement frequency from its multi-ancestor noise pattern. The lowest 11% rate covers CHAIN-5, HIVEPIDEMIOLOGY, and ROSENBROCK, where the initial skeleton is already close to the final one after

$n_{\text{obs}} = 100$ samples (CHAIN-5/HIVEPIDEMIOLOGY) or the SCM is synthetic and well-conditioned (ROSENBROCK).

E.3 Update-Frequency Sensitivity

We sweep $T_{\text{update}} \in \{3, 5, 10\}$ on the full ten-benchmark suite at $T = 50$, holding all other settings fixed. Table 18 reports mean $\pm$ std of $Y^\star$ at $n = 20$ seeds per cell.

Table 18: Update-frequency sensitivity for AGENTCBO-PRIOR, mean $\pm$ std of $Y^\star$ over 20 seeds. The default $T_{\text{update}} = 5$ corresponds to up to 10 rediscovery rounds at $T = 50$.

Benchmark	$T_{\text{update}} = 3$	$T_{\text{update}} = 5$ (default)	$T_{\text{update}} = 10$
COMPLETEGRAPH	-2.52 ± 1.42	-2.50 ± 1.21	-2.35 ± 1.45
TOYGRAPH	-2.11 ± 0.17	-2.11 ± 0.17	-1.99 ± 0.15
PSAGRAPH	$+5.18 \pm 0.06$	$+5.18 \pm 0.07$	$+5.21 \pm 0.06$
CCBOHEALTH	$+4.96 \pm 0.09$	$+4.96 \pm 0.09$	$+4.99 \pm 0.09$
HIVEPIDEMIOLOGY	-1.50 ± 0.01	-1.50 ± 0.01	-1.50 ± 0.01
CHAIN-5	-1.32 ± 0.01	-1.32 ± 0.01	-1.30 ± 0.01
ROSENBROCK	-90.98 ± 0.05	-90.98 ± 0.05	-89.21 ± 0.07
ACKLEY	-7.51 ± 0.01	-7.51 ± 0.01	-7.50 ± 0.01
PREDATORPREY	-2.31 ± 0.01	-2.31 ± 0.01	-2.29 ± 0.01
PISHCATGRAPH	$+0.28 \pm 0.10$	$+0.30 \pm 0.10$	$+0.42 \pm 0.17$

The default $T_{\text{update}} = 5$ (up to 10 rediscovery rounds) and $T_{\text{update}} = 3$ (up to roughly 16 rounds) produce similar endpoint means in Table 18. Increasing the interval to 10 (up to five rounds) weakens or ties the displayed means; the largest change is on ROSENBROCK, from -90.98 to -89.21.

This sweep measures LP rediscovery and the outer commit gate in AGENTCBO-PRIOR.

We retain $T_{\text{update}} = 5$ as the default, giving similar reported endpoints to interval 3 with fewer rediscovery rounds, while retaining more update opportunities than interval 10.

E.4 Threshold-Value Rationale

The two confidence thresholds $\tau_{\text{req}} = 0.7$ and $\tau_{\text{sup}} = 0.8$ were fixed before the main experiments on the basis of the following argument. At $\tau_{\text{req}} = 0.7$ we demand LLM agreement on both directions of an edge, i.e., $P_{ij} > 0.7$ and $P_{ji} < 0.3$, before writing it into PC's background-knowledge set $\mathcal{B}$; this is a hard constraint, so an incorrect LLM edge can distort PC's orientation or skeleton search. We therefore require near-unanimity in both directions. At $\tau_{\text{sup}} = 0.8$ we admit edges that PC dropped under finite-sample noise, subject to a relaxed CI test at level 3α; the strictly higher threshold for this re-admission, relative to τ_{req}, prevents the LLM from over-riding a statistically dropped edge unless its confidence is clearly above the hard-constraint bar. A broader sweep over $\tau_{\text{req}} \in \{0.6, 0.7, 0.8\}$ and $\tau_{\text{sup}} \in \{0.7, 0.8, 0.9\}$ is left to future work; the current paper keeps these thresholds fixed and evaluates the rest of the pipeline around that operating point.

For the guarded refinement rule (Eq. (6)), all main experiments use $\Delta_1 = \Delta_2 = 0$: noise robustness is provided by the holdout-NLL term in Eq. (7), while routing-level selectivity is imposed by the outer COMMIT GATE (whose acceptance rates of 11–33% across the ten benchmarks are reported above). We set $m_{\max} = 5$ for residual repair on the 3–11 node benchmark suite; in our runs the cap is rarely binding because the residual set $\mathcal{R}_t$ itself contains few admissible candidates after PC anchoring and target-relevance filtering.

E.5 LLM Size Robustness

We replicated the LP engine on four representative benchmarks across three Qwen3.5 sizes (35B, 122B, 397B), holding all other hyperparameters fixed. Each cell is mean$\pm$std over $n = 20$ seeds. Table 19 reports similar endpoint values across these sizes, providing a within-family comparison of model-size sensitivity.

Table 19: LLM size comparison for AGENTCBO-PRIOR. Mean±std of best $Y^\star$ over $n = 20$ seeds on four minimization benchmarks within the Qwen3.5 family.

Benchmark	Qwen3.5-35B	Qwen3.5-122B	Qwen3.5-397B
TOYGRAPH	-2.11 ± 0.17	-2.11 ± 0.17	-2.11 ± 0.17
CHAIN-5	-1.32 ± 0.01	-1.32 ± 0.01	-1.30 ± 0.01
COMPLETEGRAPH	-2.50 ± 1.21	-2.56 ± 0.68	-2.58 ± 0.97
PSAGRAPH	$+5.18 \pm 0.07$	$+5.20 \pm 0.08$	$+5.25 \pm 0.18$

Table 20: Component contracts in AGENTCBO. Inputs and outputs distinguish LLM messages from statistical-engine and optimizer state. Activation identifies when a component is used, while authority specifies the changes it can make.

Component and role	Input	Output	Activation	Authority
LP PAIRWISE LLM Semantic prior elicitation	Variable pair, names, and domain context	Cause, effect, direction, confidence, and rationale	Prior elicitation	Supplies pairwise causal judgments
LP GRAPH ENGINE Observation-anchored graph construction	Pairwise judgments, observational data, and variables	Proposal DAG, frozen backbone, residual pairs, and confidence	Stack initialization and scheduled rediscovery	Constructs proposals through PC anchoring and guarded supplementation
CALIBRATION AGENT Statistical discovery configuration	LLM: sample and variable counts, names, approximate linearity and Gaussianity, and domain context. Pipeline: observations and LP state	LLM: algorithm/hyperparameter JSON. Pipeline: statistical challenger DAG	Calib/Full, after LP	Pipeline imports acyclic, score-improving residual edits while preserving the backbone
RESIDUAL REPAIR AGENT Bounded local refinement	Current graph, PC adjacency, tool descriptions, state summary, and repair history. Observational evidence is accessible only through tools	Diagnostic actions and local-edit proposals	Full, after calibration	Pipeline imports at most five acyclic, score-improving residual edits while preserving the backbone
COMMIT GATE Graph replacement	Committed and candidate graphs, confidence, manipulators, and target	Retained or replacement graph	Periodic rediscovery	Applies confidence, manipulative-ancestry, and ancestor-subgraph nSHD checks
CBO RUNTIME Route construction and intervention optimization	Committed graph, manipulators, target, observations, archived interventions, domains, and costs	RuntimeES, do-priors, selected interventions, and updated surrogates	Initialization, observe/intervene steps, and graph changes	Selects intervention sets and values by cost-normalized CEI
CROSS-GRAPH ARCHIVE Intervention reuse	ES-keyed intervention tuples and active ESs	Stored tuples and compatible per-ES training data	Intervention recording, graph changes, and ES re-admission	Preserves intervention history and restores compatible records

F LLM Interface and Input–Output Specification

This appendix specifies the language-model interface used by AGENTCBO. All LLM components are served by the model and decoding settings described in Section 5.1; this appendix focuses on the input–output contract. The LLM is not used as an autonomous optimizer or as an unrestricted causal-discovery oracle. Instead, each LLM component receives restricted information, returns a structured proposal, and is embedded inside the guarded CBO loop. Interventional outcomes are never provided to the LLM, and the LLM never selects interventions. Intervention selection remains governed by the CBO acquisition rule, while intervention outcomes update only the per-ES surrogate and the cross-graph archive.

Module-level input–output contract. Table 20 summarizes the components, their activation conditions, and their decision authority. It distinguishes information visible to each LLM from the state used by the surrounding statistical engine and optimizer.

Intervention sets, intervention values, acquisition points, and stopping decisions remain under the BO layer's authority. The stack supplies graph proposals, while the Commit Gate controls graph replacements during rediscovery.

Parsed outputs. The LANGUAGE-PRIOR AGENT outputs a pairwise causal judgment whose algorithmically used fields are cause, effect, direction, and confidence; the reasoning field is retained for auditability but is not passed to the acquisition function. The CALIBRATION AGENT outputs algorithm-selection or hyperparameter JSON used to configure statistical discovery routines. The RESIDUAL REPAIR AGENT outputs ReAct repair steps; final graph proposals may include an adjacency matrix, variable order, confidence, and reasoning, but these proposals remain subject to downstream graph checks and commit gating. Pairwise rationales, ReAct thoughts, and textual explanations are not passed to the acquisition function. The BO policy receives only the runtime exploration sets and graph-derived surrogate priors induced by the committed routing graph.

Representative Language-Prior input and output. We include one representative pairwise call from the LANGUAGE-PRIOR AGENT on HIVEPIDEMIOLOGY, in which the domain context lists five variables but does not state any causal direction; the agent must infer directions from medical domain knowledge external to the prompt.

Input. The LLM is instructed as a causal reasoning expert. The domain context is: "HIV treatment benchmark with behavioral risk, treatment intensity, viral dynamics, drug resistance, and an aggregate health outcome." The queried variable pair is $A = T$ (treatment intensity) and $B = L$ (viral load). The model is asked to determine the most likely causal relationship between A and B, considering direct causation, reverse causation, and the possibility of no direct link. The requested output fields are cause, effect, direction, confidence, and reasoning.

Output. The returned structured judgment is: cause $= T$, effect $= L$, direction $= A \to B$, confidence $= 0.92$. The returned reasoning is: "Antiretroviral treatment intensity acts on viral replication, so increasing treatment is expected to suppress viral load rather than the reverse; the domain context does not state the direction explicitly, but the standard mechanism in HIV therapy is treatment $\to$ viral load reduction."

The domain context names the five variables but does not state the direction $T \to L$; the agent's judgement is grounded in medical domain knowledge external to the prompt. The resulting confidence is treated as a causal-prior signal for the graph-belief construction engine.

Information excluded from the LLM. No LLM component receives interventional outcomes, the current best intervention value, the best observed target value, acquisition values, acquisition maximizers, the true benchmark graph, oracle exploration sets, or oracle causal effects. This exclusion enforces the intended separation of roles: LLMs provide causal-belief proposals, while Bayesian optimization remains responsible for intervention allocation, surrogate updating, and acquisition optimization.

Domain context and prior information. The pairwise LP call receives variable names and domain context; observational samples, realized adjacency matrices, SEM equations, benchmark code, and optimization outcomes remain outside its input. The TOYGRAPH and COMPLETEGRAPH descriptions include directional clues, and the latter also identifies manipulators. CC receives dataset-level diagnostics, while RA can access the current graph and observational evidence through tools, as specified in Table 20. ALARM retains its published variable names, and ER explicitly supplies the lower-to-higher generation order without the realized adjacency. LP-Anon and LP-Shuffled assess sensitivity to the supplied lexical information.

G Fixed-Update Routing Guarantees and Proofs

This appendix formalizes the optimizer-facing consequences of routing-equivalence. Lemma 1 gives the exact sufficient condition; Proposition 1 quantifies the effect of nonzero predictive error. Neither result requires the maintained graph to equal the data-generating graph.

G.1 Matched-State Assumptions and the Routing Interface

Fix an intervention decision t and an admissible non-routing state $Z_t = z$ before route-conditioned surrogate rebuilding. This state contains observational data, ES-indexed interventional data and their archive, intervention domains and costs, the incumbent-construction rule, and the fitting and numerical acquisition configurations. It does not contain the fitted posterior being compared. Earlier graph choices may have affected these data; the result conditions on the same realized state rather than comparing independently evolving histories.

For a candidate routing graph $\mathcal{G}$, define

$$R_{t,z}(\mathcal{G}) = \left(\widehat{\mathcal{E}}(\mathcal{G}), \{\mathcal{P}_{S,\mathcal{G}}\}_{S\in\widehat{\mathcal{E}}(\mathcal{G})}\right), \tag{11}$$

where $\mathcal{P}_{S,\mathcal{G}}$ contains all graph-derived prior objects consumed by the per-ES surrogate builder. These include the do-prior mean $\mu^{\text{prior}}_{S,\mathcal{G}}$ and variance $v^{\text{prior}}_{S,\mathcal{G}}$ used to form the causal mean and covariance. Any additional graph-dependent covariance input must also be included. Equality of $\mathcal{P}$ is pointwise on every legal training/query input to the builder, not merely on a diagnostic grid. If prior construction is randomized, we condition on its realized random inputs; an unconditional conclusion additionally requires route equality almost surely under a common coupling of that randomness.

We impose the following matched-state conditions for Lemma 1.

(A1) The ES families are finite and nonempty. Each ES is a labeled set with a nonempty intervention domain; coordinates and ES enumeration use a common canonical ordering.

(A2) The same measurable surrogate builder and numerical selector are used. All their graph dependence factors through $R_{t,z}(\mathcal{G})$. Data retrieval, prior evaluation, hyperparameter fitting, incumbent construction, and selection follow the same rules and configuration.

(A3) The downstream fitting, solver, and tie-breaking random inputs $\xi = (\xi_{\text{fit}}, \xi_{\text{sel}})$ have a common graph-independent law conditional on z. Equal routes use the same realization of these inputs. A common global seed alone is not sufficient if graph-dependent earlier calls consume different portions of a random stream.

All maps are assumed well-defined on the states considered. The resulting factorization is

$$\begin{aligned}(b_{\mathcal{G}}, m_{\mathcal{G}}, s_{\mathcal{G}}) &= \text{Build}(z, R_{t,z}(\mathcal{G}); \xi_{\text{fit}}),\\ A_{\mathcal{G}}(z,\xi) &= \text{Select}\big(z, R_{t,z}(\mathcal{G}), b_{\mathcal{G}}, m_{\mathcal{G}}, s_{\mathcal{G}}; \xi_{\text{sel}}\big).\end{aligned} \tag{12}$$

Here $b_{\mathcal{G}}$ is the incumbent entering EI, and $m_{\mathcal{G}}, s_{\mathcal{G}}$ are the predictive mean and standard deviation actually supplied to acquisition. The selector can be numerical; Lemma 1 does not assume exact global maximization.

G.2 Proof of Exact Selector Sufficiency

Proof of Lemma 1. Fix z and graphs $\mathcal{G}_1, \mathcal{G}_2$ satisfying $R_{t,z}(\mathcal{G}_1) = R_{t,z}(\mathcal{G}_2)$. Their labeled domains, archived data, prior objects, fitting configurations, and coupled fitting random inputs coincide. By the common builder in Eq. (12), they therefore produce identical b, m, s, hence the same CEI surface. Applying the same measurable numerical selector with common ordering and selection/tie-breaking random inputs yields

$$A_{\mathcal{G}_1}(z,\xi) = A_{\mathcal{G}_2}(z,\xi) \quad \text{for almost every } \xi. \tag{13}$$

Their next-intervention laws at this fixed state consequently agree. No existence or uniqueness of a global maximizer is needed. If route equality holds for every admissible z, this equality holds at every such state; equality at one state alone makes no claim about other states. □

A known-graph CBO route can serve as the reference under the same state and builder. In fully observed settings this can be the route of $\mathcal{G}^\star$. With latent variables, the reference must undergo the same observable-variable processing and be a valid input to the same builder; a full latent DAG is not automatically interchangeable with the maintained observed-variable proxy. The result preserves a reference selector, not identification of the true interventional law. Route equality is sufficient, not necessary, since distinct acquisition surfaces can select the same intervention.

G.3 Proof of Approximate CEI Stability

Let $\mathcal{E}$ be the common finite nonempty labeled family and $\mathcal{A} = \bigcup_{S\in\mathcal{E}}\{S\} \times \mathcal{D}(S)$ the common action space. Fix the two realized rebuilt predictors. For minimization, write $g(a) = b - m(a)$ and $\widetilde{g}(a) = \widetilde{b} - \widetilde{m}(a)$, with real-valued means and incumbents and finite standard deviations $s(a), \widetilde{s}(a) \geq 0$. Assume common finite costs $c(a) \geq c_{\min} > 0$ and the finite uniform bounds in Proposition 1. Define

$$I(g,s) = \mathbb{E}[(g+sU)_+], \qquad \alpha(a) = \frac{I(g(a),s(a))}{c(a)}, \qquad U \sim \mathcal{N}(0,1), \tag{14}$$

and define $\widetilde{\alpha}$ analogously. Thus α here denotes the complete cost-normalized acquisition in Eq. (9). For maximization the same result holds with $g = m - b$.

Proof of Proposition 1. For $s > 0$, the Gaussian integral gives $I(g,s) = g\Phi(g/s) + s\phi(g/s)$, where Φ, ϕ are the standard normal CDF and density. Differentiating this closed form yields $\partial_g I = \Phi(g/s)$ and $\partial_s I = \phi(g/s)$. Their absolute values are bounded by 1 and $1/\sqrt{2\pi}$, respectively. Integrating along the two coordinate directions gives

$$|I(g,s) - I(\widetilde{g},\widetilde{s})| \leq |g-\widetilde{g}| + \frac{|s-\widetilde{s}|}{\sqrt{2\pi}}. \tag{15}$$

This also holds when either standard deviation is zero. Indeed, $I(g,0) = g_+$ and the positive-part map is 1-Lipschitz, so $|I(g,s) - I(g,0)| \leq s\,\mathbb{E}|U| \to 0$. Continuity extends the inequality to the boundary without differentiating there. Dividing by the common $c(a) \geq c_{\min}$ and taking the supremum over $\mathcal{A}$ proves Eq. (3).

Let the perturbed selector return $\widetilde{a} \in \mathcal{A}$ with $\widetilde{\alpha}(\widetilde{a}) \geq \sup_{a\in\mathcal{A}} \widetilde{\alpha}(a) - \zeta$ for finite $\zeta \geq 0$. Since its value is finite, this condition entails a finite perturbed supremum; the uniform bound then also makes the reference supremum finite. For every $a \in \mathcal{A}$,

$$\alpha(a) \leq \widetilde{\alpha}(a) + \eta \leq \widetilde{\alpha}(\widetilde{a}) + \zeta + \eta \leq \alpha(\widetilde{a}) + 2\eta + \zeta. \tag{16}$$

Taking the supremum over a proves the reference-acquisition loss bound, even if the reference supremum is not attained. □

Corollary 1 (Preservation of the selected ES)**.** *Under Proposition 1, let* $M_S = \sup_{\mathbf{x}\in\mathcal{D}(S)} \alpha(S,\mathbf{x})$ *be finite and have a unique best ES* $S^\star$*. For* $|\mathcal{E}| > 1$ *define*

$$\Gamma = M_{S^\star} - \max_{S\neq S^\star} M_S, \tag{17}$$

and set $\Gamma = +\infty$ *when* $|\mathcal{E}| = 1$*. If* $\Gamma > 2\eta + \zeta$*, every* ζ*-optimal perturbed selection uses* $S^\star$*.*

Proof. The singleton-family case is immediate. Otherwise a returned $\widetilde{a} = (S,\mathbf{x})$ with $S \neq S^\star$ would have reference-acquisition loss at least $M_{S^\star} - M_S \geq \Gamma > 2\eta + \zeta$, contradicting Proposition 1. This preserves the ES identity, not necessarily the continuous intervention value within it. □

G.4 Why Both Routing Components Matter

The common-family condition cannot be replaced by agreement only on shared actions. Consider $\mathcal{E} = \{\{X_1\},\{X_2\},\{X_1,X_2\}\}$ and $\widetilde{\mathcal{E}} = \{\{X_1\},\{X_2\}\}$, each with singleton domains and unit costs. For any $q, L > 0$, take deterministic Gaussian predictions ($s = 0$), incumbent $b = 0$, and means $-q$ on the two singleton actions and $-(q+L)$ on the joint action. The shared predictions agree exactly, but removing the joint ES incurs reference-CEI loss L. This is an optimizer-interface counterexample, not an assertion that every such predictive assignment is induced by a particular fitted SCM.

Conversely, equal ES families alone do not preserve selection. On two actions with unit cost, $b = 0$, and $s = 0$, exchanging the means -1 and -2 reverses the maximizing action without changing the family. Uncertainty matters as well: when $g = 0$, EI is $s/\sqrt{2\pi}$, so matching prior means alone cannot bound CEI differences. Together these examples explain why routing analysis tracks both the available sets and their predictive scoring.

Relation to structural and prior-work criteria. Graph equality is stronger than route equality at matched inputs. Structural discrepancies outside the inputs used by the routing builder can be invisible to CEI, while a single edit removing an available joint intervention can affect selection. Thus structural distance alone does not supply the predictive-error bounds required above. Section 3 discusses the corresponding prior-work distinctions. Our analysis concerns the joint graph-dependent RuntimeES and rebuilt predictive interface of committed-graph CBO, with an explicit CEI perturbation and ES-margin bound.

Applying the fixed-update guarantees. Applying Proposition 1 requires uniform error bounds on the rebuilt predictive quantities, including incumbent changes and uncertainty, and a bound ζ on numerical maximization error. These are analytic conditions on the predictor and selector. The implemented ancestry and nSHD checks provide graph-commitment proxies, while the fitted-prior-mean diagnostic measures one component of predictive change. Extending this analysis to evolving optimization runs requires controlling subsequent data, refits, archives, and accepted updates.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Sections 4.2–5 support the architecture, conditional theory, and empirical claims. Appendices B.1 and C provide the matched controls, replay, corruption study, and larger-graph evidence, with protocol-specific conclusions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix A.1 discusses the fixed-update scope, prior exposure and supplied ER ordering, single-family LLM evaluation, prescribed update-time corruption, and task-dependent refinement gains.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Section 4.2 states fixed-update selector sufficiency and approximate CEI stability. Appendix G gives the complete assumptions, proofs, selected-ES margin corollary, and scope conditions.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Reproducibility is layered across Section 5.1 (protocol), Appendix B (benchmark SEMs), Appendix D (Algorithms 1–2 and module descriptions), Appendix E (hyperparameter sensitivity), and Appendix F (LLM input–output contract).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The core AgentCBO implementation, benchmark implementations, and usage instructions are publicly available at `https://github.com/Qiyu-Wei/AgentCBO_Official`. The release includes the three AgentCBO variants and matched dynamic PC-only and known-graph controls.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5.1 gives the main-suite protocol (twenty seeds, $T = 50$, $n_{\text{obs}} = 100$). Appendices B.1 and C specify the additional controls, corruption configuration, ALARM construction, and ER generation and aggregation. Algorithm 2 and Appendices D–F describe engine settings, sensitivity, and LLM inputs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Main-suite endpoints are mean±std over twenty seeds. Appendix C.2 specifies paired benchmark-level tests and Holm correction. Additional studies report hit rates, paired seed contrasts, and DAG-level summaries with confidence intervals; Tables 10 and 12 identify the pairing and aggregation units.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D reports main-suite worker classes and standalone engine discovery costs. Tables 13 and 14 separate ER per-run optimization time from one-time prior elicitation.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Synthetic and semi-synthetic benchmarks only, no human subjects, no personally identifiable information, and the LLM (Qwen3.5-35B-A3B Apache 2.0) is open-weights, with broader-impact discussion in Appendix A.2.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix A.2 covers both positive (reduced expensive interventions in scientific optimization) and negative (over-trust in incorrect discovered graphs in high-stakes domains) directions, framing AgentCBO as decision-support requiring human oversight.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release a new high-risk model, dataset, or scraped corpus; it studies existing LLM-assisted methods on synthetic and semi-synthetic optimization benchmarks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Open-weight components (Qwen3.5-35B-A3B and vLLM, both Apache 2.0) are credited with URLs and licenses in Appendix D; benchmark SEMs follow their original sources, cited per row in Appendix B, and baselines are credited to their original papers. The ALARM topology is not redistributed: the released code instructs users to obtain it from bnlearn under its GPL-2-or-later license.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released code (`https://github.com/Qiyu-Wei/AgentCBO_Official`, MIT License) ships with a README covering installation, vLLM configuration, running the three AgentCBO variants and matched controls, benchmark setup, code layout and tests. No new datasets or trained models are released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLM usage is central to AgentCBO's method, with the three components (LANGUAGE-PRIOR AGENT, CALIBRATION AGENT, RESIDUAL REPAIR AGENT) described in Section 4.3, the per-agent input–output contract and information barrier in Appendix F, and the model and decoding settings in Appendix D.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.