# Testing Causal Models with Hidden Variables in Polynomial Delay via Conditional Independencies

Purdue University
Columbia Engineering
Mohamed bin Zayed University of Artificial Intelligence

Hyunchai Jeong*[1], Adiba Ejaz*[2], Jin Tian[3], Elias Bareinboim[4]

[1]Purdue University [2,4]Columbia University [3]Mohamed bin Zayed University of Artificial Intelligence [1,2,4]CausalAI Lab

## Motivation

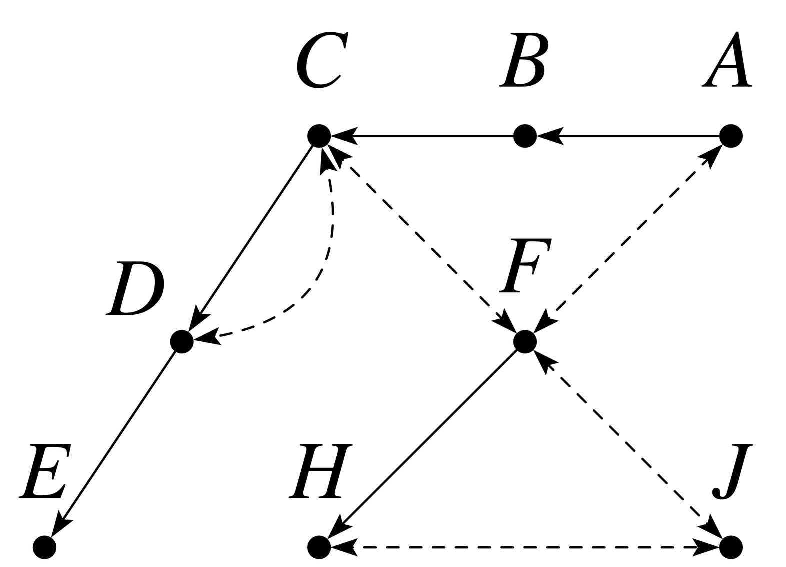
Causal models help us estimate causal effects when experiments are infeasible. But an incorrect model leads to incorrect conclusions. **How can we efficiently test causal models with hidden variables?**

## Contributions

We propose an efficient algorithm, **ListCI**, for testing causal graphs via conditional independencies (CIs):

1. **Real-world applicability:** test causal models with hidden variables against non-parametric data distributions.
2. **Fewer tests:** avoids exponentially many redundant CI tests by leveraging the c-component local Markov property.
3. **Faster execution:** runs in polynomial delay, enabling graph testing in poly-time intervals.

## An Example




- $A \to B :=$ $A$ causes $B$
- $C \leftrightarrow D :=$ $C$ and $D$ are confounded by hidden factors
- *C-Component* := bidirected connected component e.g., $\{A, C, D, F, H, J\}$

*Fig. 1. A causal graph $\mathcal{G}$*

This graph encodes **1198** CIs!

- For example, $A \perp E \mid D$ and $J \perp B,\ C \mid A,\ D,\ E$.
- Asymptotically $\Theta(4^n)$.

Testing all of them is redundant!

- A subset of only **11** CIs implies all others.

## Result I: A New Local Markov Property

We introduce the **c-component local Markov property,** which defines an exponentially smaller set of CIs that provably implies all others.

*Local Markov Property (Lauritzen et al 1990):* For every variable $X$ in $\mathcal{G}$,

$$X \perp \textit{Non-Desc}(X) \setminus pa(X) \mid pa(X)$$

> *C-Component Local Markov Property:* For every variable $X$ in $\mathcal{G}$ and every ancestral c-component $\mathbf{C}$ relative to $X$,
> $$X \perp \mathbf{S}^{+}_{\mathbf{C}} \setminus Pa(\mathbf{C}) \mid pa(\mathbf{C})$$

| Causal graphs with hidden variables? | LMP ❌ | C-LMP ✅ |
|---|---|---|

## Result II: A Poly-Delay Algorithm

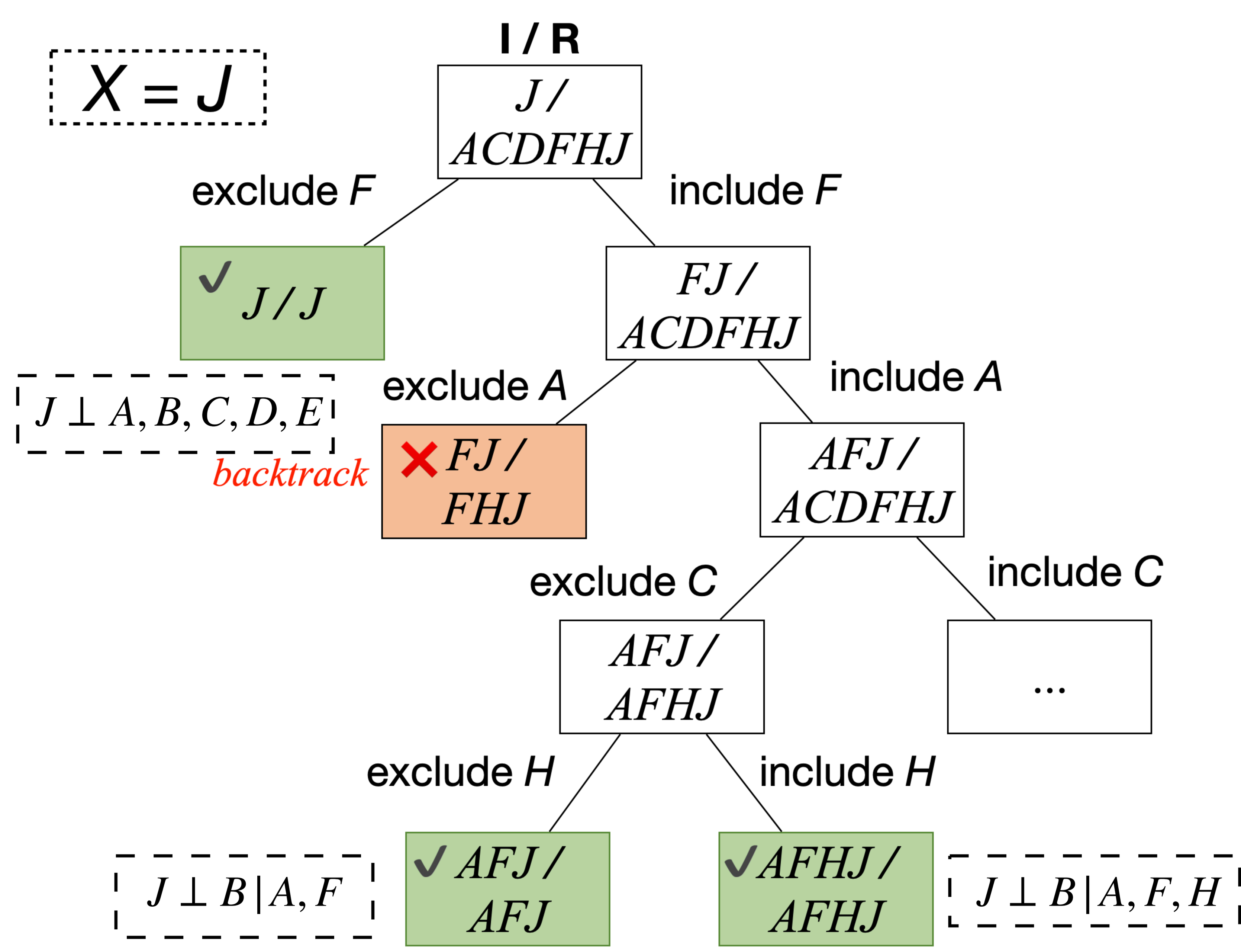



*Fig. 2. Search tree for listing CIs invoked by C-LMP for the variable $J$ in graph $\mathcal{G}$ (Fig. 1). We use a divide-and conquer approach with backtracking to enumerate ACs and output the corresponding CIs.*

## Empirical Performance

ListCI is **orders of magnitude faster than baselines** on `bnlearn` graphs with up to 100 nodes.

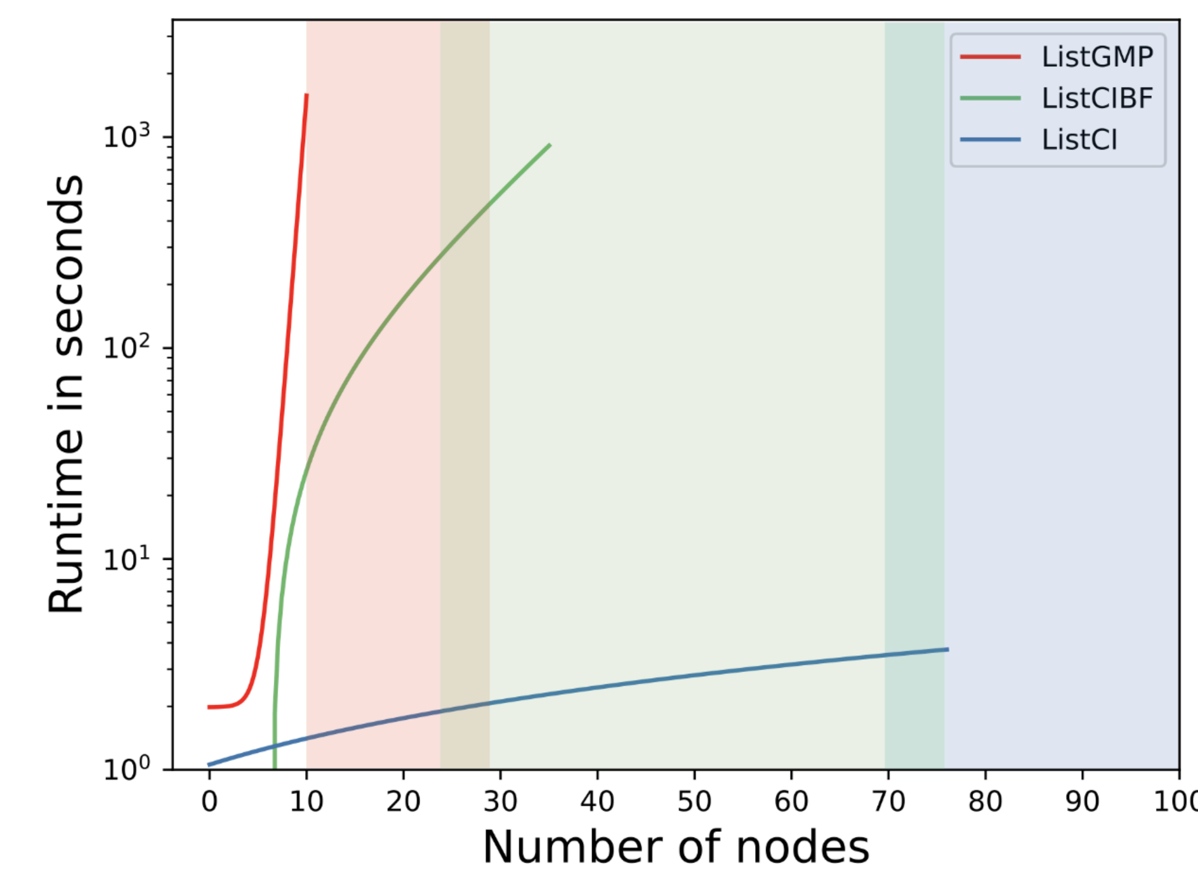



*Fig. 3. Runtime of ListCI vs. baselines on bnlearn graphs. Each shaded region indicates the interval of $n$ on which the relevant algorithm timed out on some graphs with $n$ nodes. The y-axis uses a logarithmic scale.*

## Phase Transitions?

How does the graph topology associated with c-components affect the #CIs invoked by C-LMP?

**In theory,** $\mathcal{O}(n2^s)$ where $s$ is the size of the largest c-component.

**In practice,** as we increase the number of bidirected edges:

- Phase 1: While c-components are sparse, #CI increases exponentially.
- Phase 2: As c-components become more dense, #CI decays exponentially.

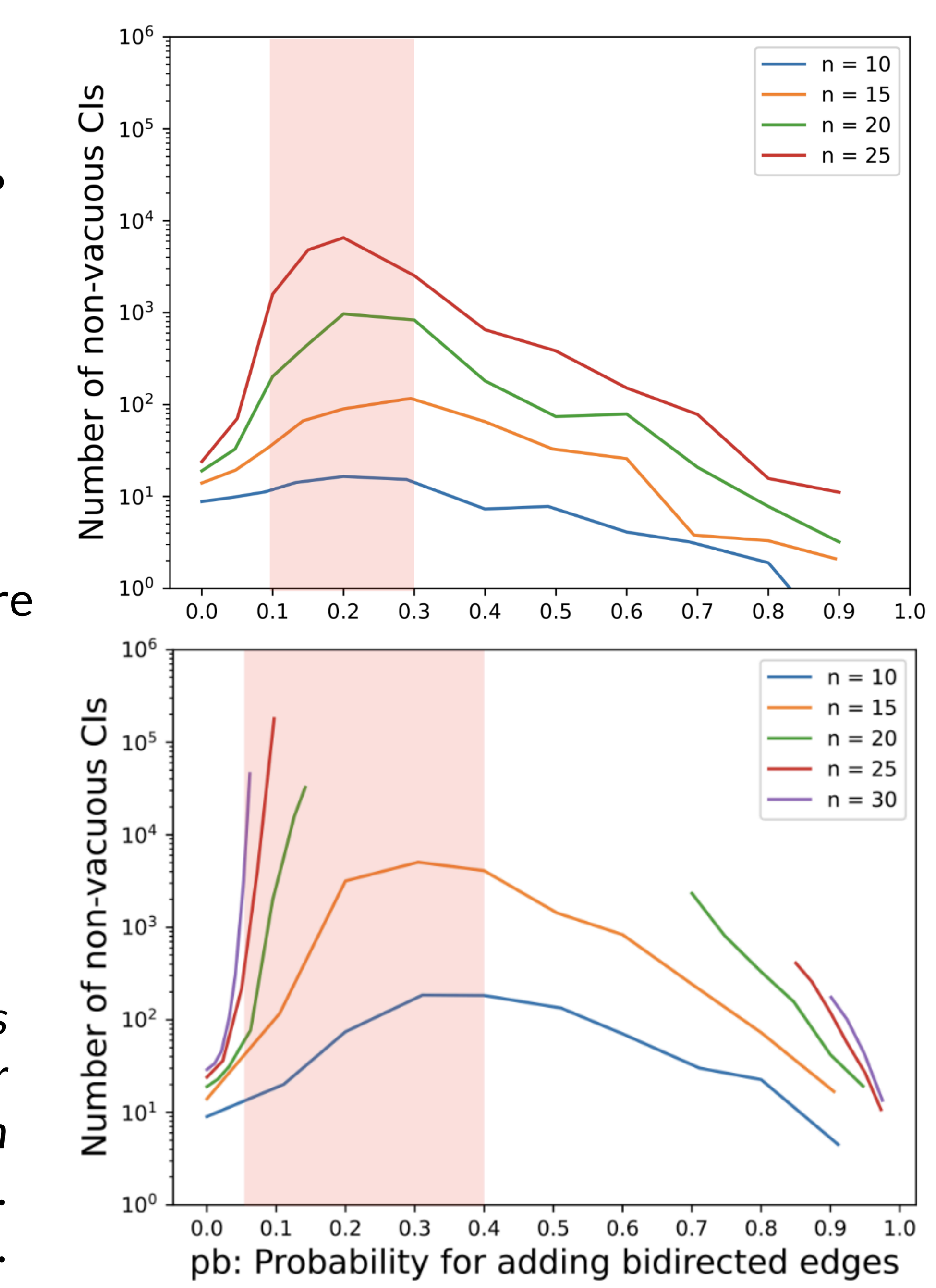



*Fig. 4. #CIs invoked by C-LMP vs probability of a bidirected edge for Erdos-Renyi random graphs with $n$ nodes. Top: no directed edges. Bottom: $n$ directed edges.*

39th Annual AAAI Conference on Artificial Intelligence, Feb 25 – March 4. 2025, Philadelphia, PA, USA