



Abstract

We propose **Sequential Non-Ancestor Pruning** (SNAP) to identify causal effects between **target variables** in a **computationally and statistically efficient way**. SNAP can be used for pre-processing, or as a standalone sound and complete causal discovery algorithm. Both approaches can substantially reduce the number of CI tests and the computation time without compromising the quality of causal effect estimations.

Problem setting

We formalize the problem of estimating causal effects between targets efficiently, without knowledge of the true causal graph.

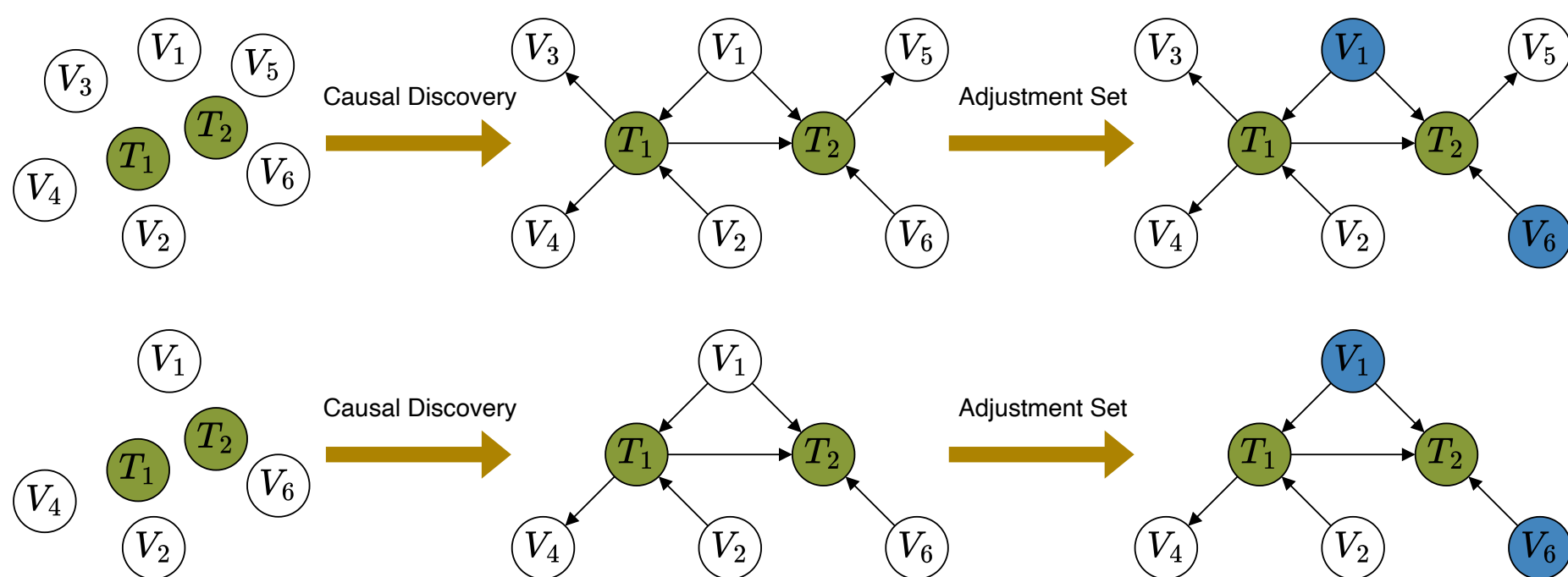
Targeted causal effect estimation with an unknown graph

Given a joint distribution p over $\mathbf{V}$ and targets $\mathbf{T} \subseteq \mathbf{V}$, we consider the task of estimating in a **computationally and statistically efficient way** the interventional distributions $P(T_i | do(T_j))$, for all pairs $T_i, T_j \in \mathbf{T}$.

- **Computationally efficient**: in terms of CI tests and computation time.
- **Statistically efficient**: optimal in terms of asymptotic variance.

We assume p is observational, Markov, faithful, and causally sufficient.

Insight: Targeted causal effect estimation with an unknown graph may not require discovering the full causal graph over all variables.



Sequential Non-Ancestor Pruning (SNAP)

In DAGs, the non-ancestors of the targets $\mathbf{T}$ are not needed as part of the statistically efficient adjustment sets between $\mathbf{T}$ [Guo et al. 2023].

We show that appropriate over-estimations of the **possible ancestors** of $\mathbf{T}$ suffice to **identify statistically efficient adjustment sets** between $\mathbf{T}$.

Lemma: Possibly Ancestral Set

Let $G(\mathbf{V})$ be the CPDAG over $\mathbf{V}$. Let $\mathbf{V}^* \subseteq \mathbf{V}$ be a possibly ancestral set, i.e. $PossAn_G(\mathbf{V}^*) \subseteq \mathbf{V}^*$, and $G(\mathbf{V}^*)$ the CPDAG over $\mathbf{V}^*$. Then $G(\mathbf{V})|_{\mathbf{V}^*} = G(\mathbf{V}^*)$.

Thus, causal discovery only on the possible ancestors $PossAn(\mathbf{T})$

- ✓ enables efficient causal effect estimation between targets $\mathbf{T}$,
- ✓ potentially requires **much fewer CI tests**,
- ✗ but requires knowledge of $PossAn(\mathbf{T})$.

Sequential Non-Ancestor Pruning progressively identifies and removes the non-ancestors of targets $\mathbf{T}$ **during** the process of causal discovery.

We implement SNAP by modifying the skeleton search of PC:

- orient v-structures after each set of CI tests at a given order i ,
- **identify and remove some definite non-ancestors** of targets $\mathbf{T}$,
- continue the search only over the remaining variables.

Algorithm 1 Sequential Non-Ancestor Pruning - SNAP(k)

Require: Vertices $\mathbf{V}$, Targets $\mathbf{T} \subseteq \mathbf{V}$, Maximum order k

- 1: $\hat{\mathbf{V}}^0 \leftarrow \mathbf{V}, \hat{\mathbf{U}}^{-1} \leftarrow$ fully connected undirected graph over $\mathbf{V}$
- 2: **for** $i \in 0..k$ **do**
- 3: $\hat{\mathbf{U}}^i \leftarrow$ induced subgraph of $\hat{\mathbf{U}}^{i-1}$ over $\hat{\mathbf{V}}^i$ ▷ Prune definite non-ancestors
- 4: $\hat{\mathbf{U}}^i \leftarrow$ Skeleton step at order i
- 5: **if** $i < 2$ **then** ▷ Orient v-structures
- 6: $\hat{\mathbf{G}}^i \leftarrow$ Orient V-structures as in PC
- 7: **else**
- 8: $\hat{\mathbf{G}}^i, \hat{\mathbf{U}}^i \leftarrow$ Orient V-structures as in RFCI
- 9: $\hat{\mathbf{V}}^{i+1} \leftarrow$ all $V \in \hat{\mathbf{V}}^i$ with a possibly directed path to any $T \in \mathbf{T}$ in $\hat{\mathbf{G}}^i$
- 10: $\hat{\mathbf{G}}^k \leftarrow$ induced subgraph of $\hat{\mathbf{G}}^k$ over $\hat{\mathbf{V}}^{k+1}$
- 11: **return** $\hat{\mathbf{V}}^{k+1}, \hat{\mathbf{G}}^k$

The SNAP(∞) algorithm

SNAP(∞) extends SNAP(k) to a **stand-alone algorithm** that

1. runs SNAP(k) until completion, i.e. $k = |\mathbf{V}| - 2$,
2. runs Meek's rules to identify definite non-ancestors one last time,
3. outputs the induced subgraph of the full CPDAG over $PossAn(\mathbf{T})$.

Theoretical results

The remaining variables returned by SNAP(k) for any order k are sufficient to identify the optimal adjustment sets between targets $\mathbf{T}$.

Theorem: SNAP(k) is sound

At each step $i = 0, \dots, k$ of SNAP(k), $\hat{\mathbf{V}}^{i+1}$ contains $PossAn(\mathbf{T})$. Additionally, $\hat{\mathbf{V}}^{i+1}$ is a possibly ancestral set, i.e. $PossAn_G(\hat{\mathbf{V}}^i) \subseteq \hat{\mathbf{V}}^{i+1}$.

This means that SNAP(k) with $k < |\mathbf{V}|$ can be used to **pre-filter** some non-ancestors before running standard causal discovery algorithms.

SNAP(∞) returns exactly the CPDAG over the possible ancestors of $\mathbf{T}$.

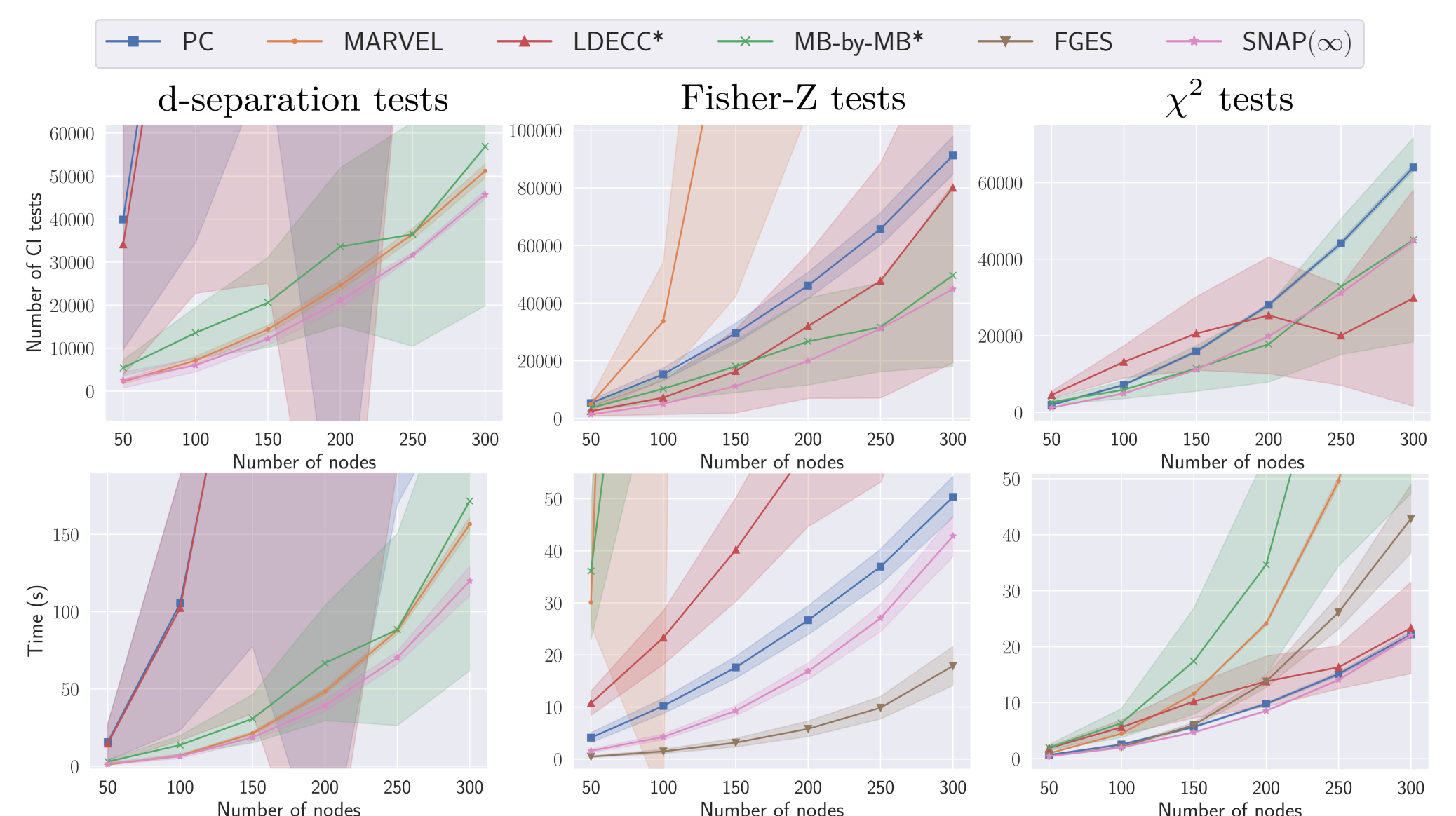
Theorem: SNAP(∞) is sound and complete

Let $\hat{\mathbf{G}}$ be the output graph of SNAP(∞) for targets $\mathbf{T}$. Then, SNAP(∞) returns $\hat{\mathbf{V}} = PossAn(\mathbf{T})$. Additionally, SNAP(∞) is sound and complete over the possible ancestors $\mathbf{T}$, i.e. $\hat{\mathbf{G}} = G|_{PossAn(\mathbf{T})}$.

Experiments

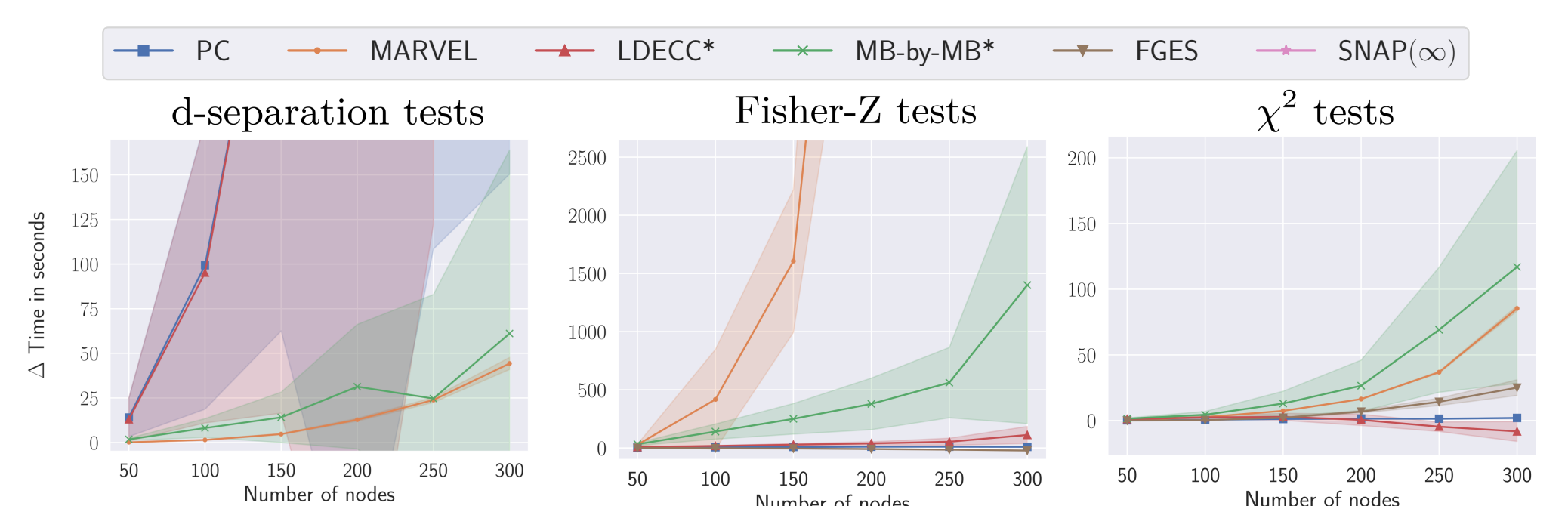
We evaluate SNAP(∞), PC, MARVEL, FGES, MB-by-MB and LDECC, and their combination with SNAP(0).

SNAP(∞) is **consistently one of the best in each domain** terms of CI tests and computation time, with a comparable intervention distance.

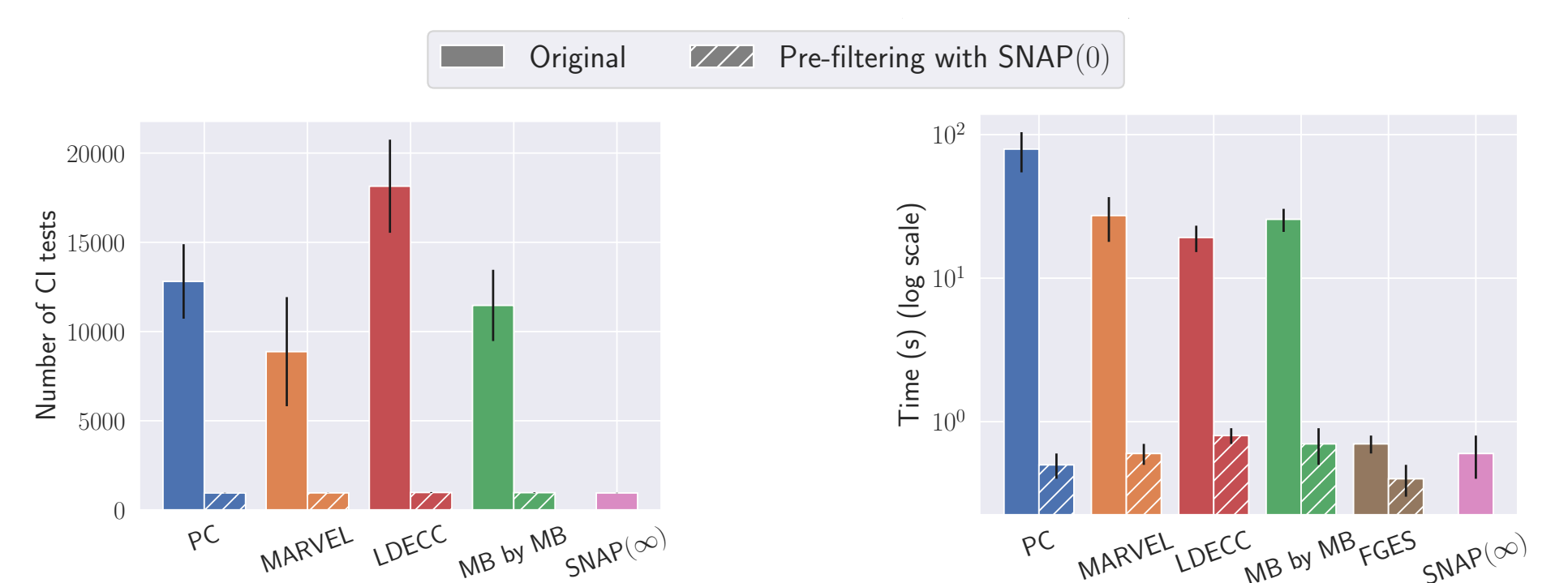


1000 synthetic samples with 4 targets, average degree of 3 and maximum degree of 10.

Pre-filtering with SNAP(0) greatly improves baselines in most settings.



Difference in time for without and with pre-filtering with SNAP(0).



1000 linear Gaussian semi-synthetic samples from the MAGIC-NIAB network with 4 targets.