Breaking Bad: Component-Wise Parent Deletion for Score-Based Causal Discovery

Min Woo Park[1*] **Taehui Yun**[1*] YoungIn Jang[1] Yoonseok Yeom[1] Jonghwan Kim[1]
Jiyeon Kang[2] Songseong Kim[2] Hyemin Jung[2] Sangmin Lee[2] Jongseong Jang [2†] Sanghack Lee[1,3†]

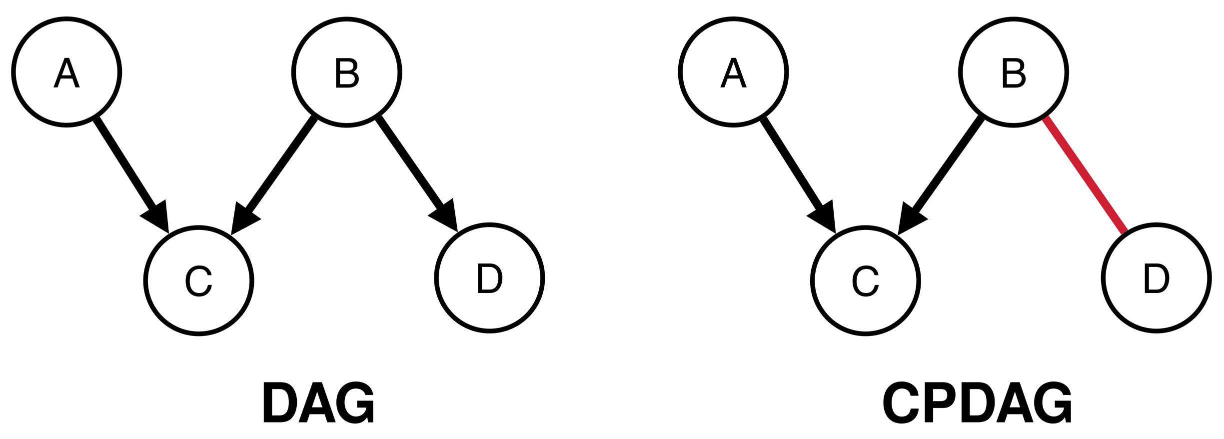
[1]Seoul National University [2]LG AI Research [3]SNU-LG AI Research Center

Seoul National University Graduate School of Data Science
LG AI Research
uai2026

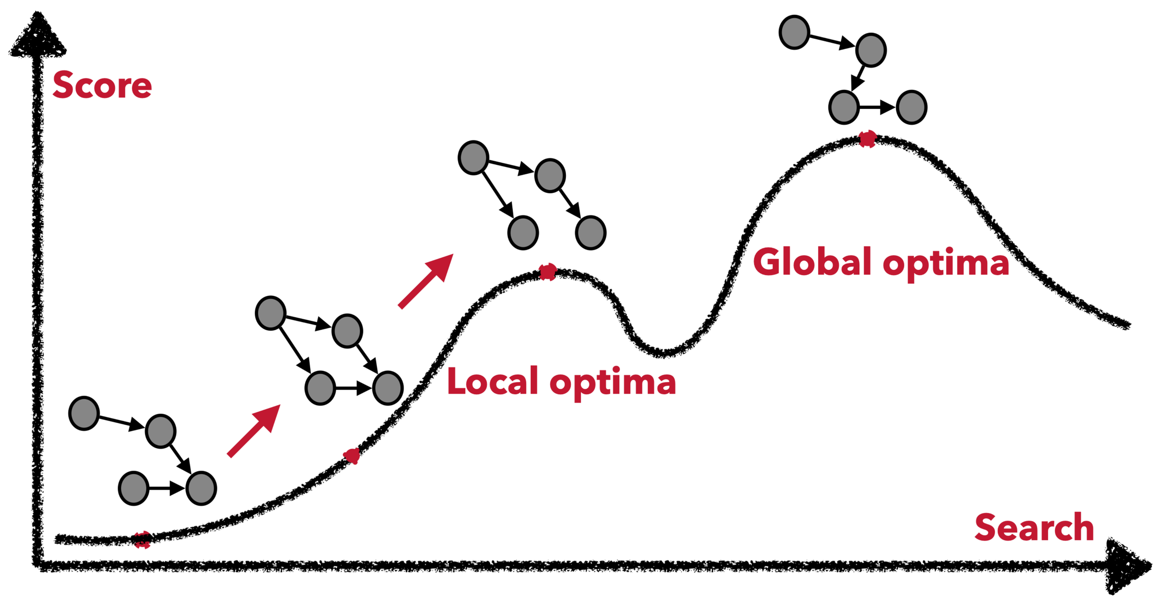
Background

Causal discovery aims to recover the underlying causal structure from data.

Markov Equivalence Class (MEC). A collection of DAGs sharing the same conditional independencies, represented by a **CPDAG**.



Score-based causal discovery learns a **CPDAG** from data that optimizes a score function (e.g., Bayesian Information Criterion; BIC).

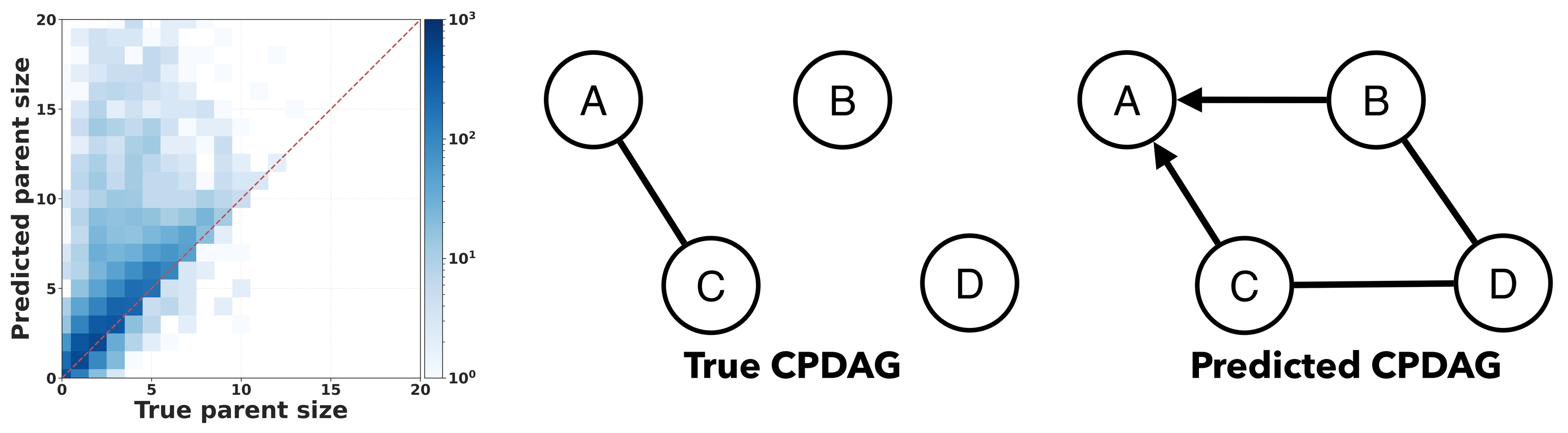


In finite samples, score-based causal discovery algorithms often become trapped in **local optima** and fail to recover the true MEC (CPDAG).

Key Observation

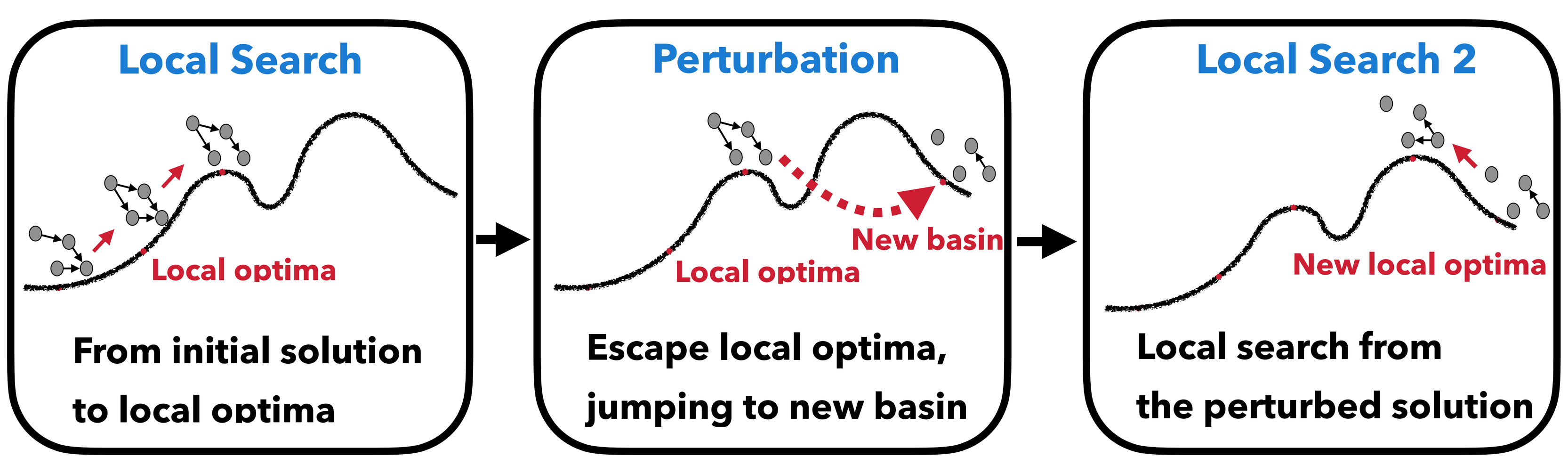
❶ **Weak scalability.** Score-based algorithms struggle in **high-dimensional** (many nodes) and **dense** (many edges) settings.

❷ **Parent set over-estimation.** Score-based algorithms tend to introduce nodes which have **excessive parent variables**, yielding overly dense CPDAGs.



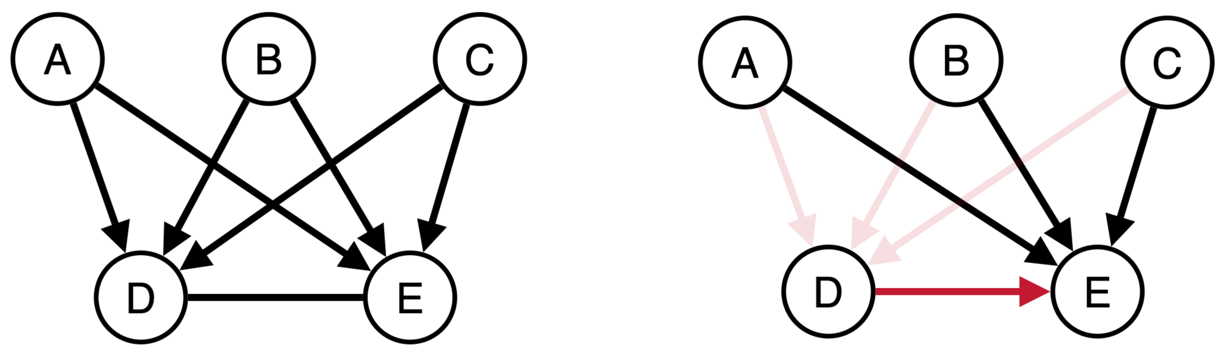
Iterated Local Search (ILS). A local search heuristic that escapes local optima by iterating: **(1)** perturb the current solution, **(2)** re-optimize, **(3)** repeat.

💡 The choice of **perturbation strategy** is the most important component.



Parent Deletion

Parent deletion. A simple perturbation strategy which **(1)** removes all incoming edges of a node and **(2)** reorients selected neighboring edges while preserving CPDAG validity.



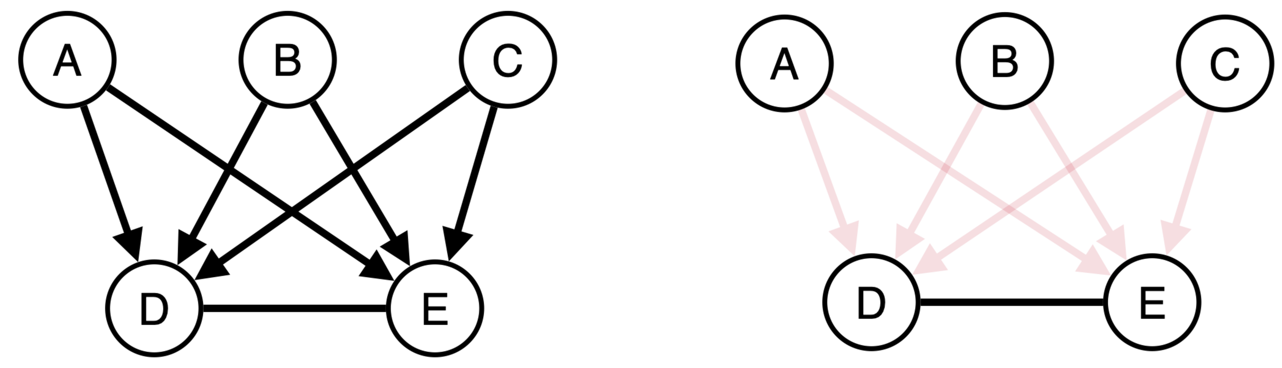
◎ **Merely deleting many parent edges leads to unintended side effects.**

Collateral parental growth. A side effect:

Nodes connected through undirected edges **share the same parent set**.
⇒ Parent deletion may increase neighboring parent sets.

Component-Wise Parent Deletion

Component-wise parent deletion. Delete parents for an **entire undirected component simultaneously.**



◎ **Simple yet powerful operator *theoretically sound*, effectively alleviating scalability issues.**

Validity. A perturbation is *valid* if it maps a **consistent extension** of the current CPDAG to one of the consistent extensions of the resulting CPDAG.

Component-wise parent deletion is **always valid** in a CPDAG;
⇒ **No additional validity checking** is required!

Structural remedy. Removing shared parents at once avoids collateral growth, aligning with our intention of **escaping dense local optima**.

Perturbation Strategy

Compatibility. Our perturbation can be **seamlessly incorporated** into existing score-based causal discovery algorithms.

Method	Perturbation
-D	Single-edge deletion
-DP	Component-wise parent deletion
DP+	Scheduling (from stronger to weaker) Component-wise → Node-wise → Edge-wise

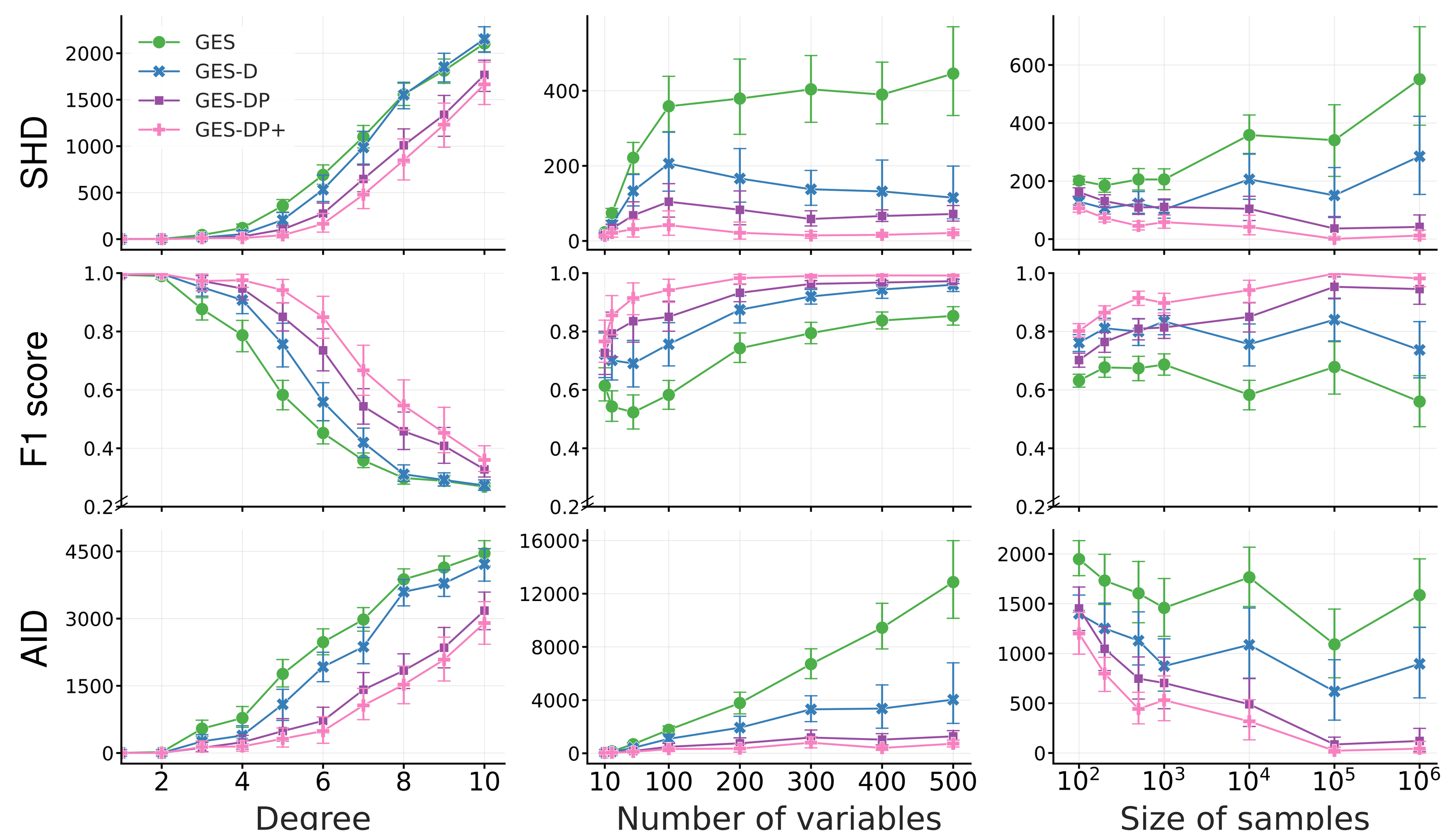
Experiments

Baselines. GES, OPS, XGES, LGES, and BOSS.

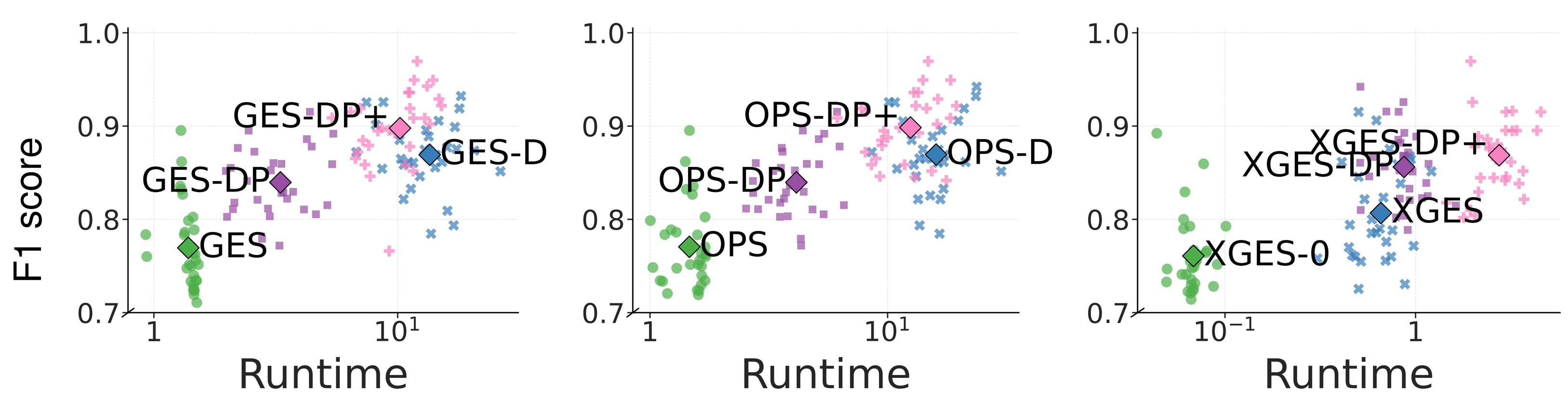
Evaluation metrics. **SHD** (↓): structural distance, **F1** (↑): edge recovery accuracy, **AID** (↓): causal effect distance, **Runtime** (↓): computational cost.

For each score-based causal discovery algorithm, we compare different perturbation strategies.

Synthetic datasets. Erdős–Rényi, Scale-Free network.



Biological networks. ARTH150, ECOLI70, MAGIC-IRRI, MAGIC-NIAB.



Conclusion

- A simple yet powerful perturbation to **escape dense local optima** in score-based causal discovery.
- Formalized single-node deletion and extended it component-wise, which **avoids collateral parental growth** without validity-checking cost.
- Notable **speedups** and substantial **performance gains** across benchmarks.

Acknowledgements. This work was supported by LG AI Research, IITP (RS-2022-II220953/50%) and NRF (RS-202300211904/50%) grants funded by the Korean government.