DCILP: A Distributed Approach for Large-Scale Causal Structure Learning

Shuyu Dong[1*], Michèle Sebag[1*], Kento Uemura[2], Akito Fujii[2], Shuang Chang[2], Yusuke Koyanagi[2], Koji Maruhashi[2]

[1]INRIA, LISN, Université Paris-Saclay, 91190, Gif-sur-Yvette, France
[2]Fujitsu Limited, Kanagawa, 211-8588, Japan
shuyu.dong@inria.fr, michele.sebag@lri.fr, {uemura.kento, fujii.akito, chang.shuang, koyanagi.yusuke, maruhashi.koji}@fujitsu.com

Abstract

Causal learning tackles the computationally demanding task of estimating causal graphs. This paper introduces a new divide-and-conquer approach for causal graph learning, called DCILP. In the divide phase, the Markov blanket $\mathbf{MB}(X_i)$ of each variable X_i is identified, and causal learning subproblems associated with each $\mathbf{MB}(X_i)$ are independently addressed in parallel. This approach benefits from a more favorable ratio between the number of data samples and the number of variables considered. In counterpart, it can be adversely affected by the presence of hidden confounders, as variables external to $\mathbf{MB}(X_i)$ might influence those within it. The reconciliation of the local causal graphs generated during the divide phase is a challenging combinatorial optimization problem, especially in large-scale applications. The main novelty of DCILP is an original formulation of this reconciliation as an integer linear programming (ILP) problem, which can be delegated and efficiently handled by an ILP solver. Through experiments on medium to large scale graphs, and comparisons with state-of-the-art methods, DCILP demonstrates significant improvements in terms of computational complexity, while preserving the learning accuracy on real-world problem and suffering at most a slight loss of accuracy on synthetic problems.

Code — https://github.com/shuyu-d/dcilp-exp
Extended version — https://arxiv.org/abs/2406.10481

1 Introduction

Discovering causal relations from observational data emerges as an important problem for artificial intelligence with fundamental and practical motivations (Pearl 2000; Peters, Janzing, and Schölkopf 2017). One notable reason is that causal models support modes of reasoning, e.g., counterfactual reasoning and algorithmic recourse (Tsirtsis et al. 2021), that are beyond the reach of correlation-based models (Peters, Bühlmann, and Meinshausen 2016; Arjovsky et al. 2019; Sauer and Geiger 2021). In the literature of causal discovery and Bayesian network learning, there are two main categories of methods, namely constraint-based methods (Spirtes et al. 2000; Meek 1995) and score function-based methods (Chickering 2002a; Loh and Bühlmann 2014) (more in Section 5). Depending on the specific method, strategies for learning large causal graphs include restricting the search space of directed graphs to that of sparse graphs (Ramsey et al. 2017; Loh and Bühlmann 2014), or transforming the underlying combinatorial problem into a continuous optimization problem (Zheng et al. 2018; Aragam, Amini, and Zhou 2019; Ng, Ghassami, and Zhang 2020; Ng et al. 2021; Lopez et al. 2022). While these strategies have resulted in significant improvements in reducing the complexity, their scalability is still limited when the number of variables and/or the degree of the sought causal graph are high.

To better tackle the computational challenges in learning large causal structures, a growing number of works consider breaking down the large-scale causal discovery problem into smaller ones defined from subsets of variables, and conducting a divide-and-conquer strategy. Such subsets of variables might be incrementally built and refined (Gao, Fadnis, and Campbell 2017); they can be based on hierarchical clustering (Gu and Zhou 2020), recursive decomposition based on conditional independence tests (Zhang et al. 2020), or through the Markov blanket (defined in Section 2) associated with each variable (Tsamardinos et al. 2003; Wu et al. 2020, 2022, 2023; Mokhtarian et al. 2021). A main challenge lies in the conquer step for the fusion or reconciliation of the partial solutions identified in the divide step; most conquer approaches are rule-based which limits their applicability.

In this paper we present the original *Divide-and-Conquer causal modelling with Integer Linear programming* approach (DCILP) to address the scalability challenge inherent to causal discovery. Formally, DCILP consists of three phases:

- Phase-1 estimates the Markov blankets $\mathbf{MB}(X_i)$ associated with each observed variable X_i, that are used to divide the causal discovery problem into smaller subproblems;
- Phase-2 independently tackles each causal subproblem defined by the variables in $\mathbf{MB}(X_i)$. Note that these causal subproblems offer a more favorable ratio between the number of samples and the number of variables; in counterpart, they may no longer satisfy the assumption of causal sufficiency (section 2) — as variables external to a Markov blanket might act as hidden confounders,

*These authors contributed equally.

having an impact on the variables within it;

- Phase-3 is a conquer phase, reconciling the causal relations identified for each subproblem into a globally consistent causal graph.

The original contribution of the proposed DCILP is twofold. Firstly, Phase-2 is parallelizable by design; the fact that it can handle the causal discovery subproblems associated with each Markov blanket, allows it to scale up to a few thousand variables. The causal insufficiency issue is mitigated by only retaining the causal relations involving the center variable of the Markov Blanket. Secondly, and most importantly, we show that the reconciliation of the causal subgraphs learned in Phase-2 can be formulated and efficiently solved as an integer linear programming (ILP) problem. Binary ILP variables are defined to represent the causal relations (causes, effects, spouses, and v-structures); logical constraints are defined to enforce their consistency, and the optimization of the ILP variables aims to find a causal graph as close as possible to the concatenation of all local subgraphs, subject to the consistency constraints. The resolution of this ILP problem can be delegated to highly efficient ILP solvers.

Overall, DCILP defines a flexible framework where different algorithmic components can be used in each phase: (i) For the Markov blanket discovery task in Phase-1, we restrict ourselves to linear structural equation models (Section 2) following the setting of (Loh and Bühlmann 2014);[1] (ii) for the causal discovery subproblems in Phase-2, we consider GES (Chickering 2002b) and DAGMA (Bello, Aragam, and Ravikumar 2022) as they are two representative and efficient state-of-the-art algorithms for causal modeling; (iii) for Phase-3, we use the Gurobi ILP solver (Gurobi Optimization 2025).

The paper is organized as follows. After presenting the formal background in Section 2, we describe DCILP in Section 3. Section 4 presents the experimental setting and results of DCILP. Section 5 discusses the position of DCILP with respect to the related work. Section 6 concludes the paper and presents some perspectives for further work.

2 Formal Background

Definition 1. Let $\mathbf{X} = (X_1, \ldots, X_d)$ be a set of d observed random variables. The linear Structural Equation Model (SEM) of $\mathbf{X}$ is a set of d equations:

$$\forall 1 \leq i \leq d, \quad X_i = \beta_{1,i}X_1 + \cdots + \beta_{d,i}X_d + \epsilon_i$$

with ϵ_i an external random variable, independent of any X_j for $j \neq i$. Coefficient $\beta_{i,i} = 0$; at most one of $\beta_{i,j}$ and $\beta_{j,i}$ is nonzero. If $\beta_{j,i} \neq 0$, X_j is said to be a cause, or parent, of X_i; X_i is said to be an effect of X_j.

The graph $G := (\mathbf{X}, E)$ with adjacency matrix $B = (\beta_{i,j})$, referred to as causal graph, is the directed graph such that the edge set E is the set of pairs (i, j) with $\beta_{i,j} \neq 0$. The directed graph of a linear SEM is usually required to be acyclic (DAG).

[1]The case of non-linear causal relations is left for further work, using for instance feature selection to elicit Markov blankets (Tsamardinos et al. 2003).

Definition 2. Consider a DAG $G = (\mathbf{X}, B)$ defined on $\mathbf{X} = (X_1, \ldots, X_d)$. The Markov blanket of variable X_i, denoted as $\mathbf{MB}(X_i)$, is the smallest set $M \subset \mathbf{X}$ such that

$$X \perp\!\!\!\perp_G \mathbf{X} \backslash (M \cup \{X_i\}) \text{ given } M,$$

where $\perp\!\!\!\perp_G$ denotes d-separation (Peters, Janzing, and Schölkopf 2017, Definition 6.1, Definition 6.26).

Causal learning commonly involves two assumptions. The *Markov property* states that the joint distribution of $\mathbf{X}$ exactly reflects the dependencies and independence relations in causal graph G, implying that $P(\mathbf{X}) = \prod_{i=1}^{d} P(X_i | \operatorname{Pa}(X_i))$ with $\operatorname{Pa}(X_i)$ denoting the set of variables parent of X_i in G. The *Markov sufficiency* assumption states that all confounders (variables causes of at least two observed variables) are also observed.

Under these assumptions, the Markov blanket $\mathbf{MB}(X_i)$ is the smallest set M such that X_i is independent of all other variables in $\mathbf{X}$ given M; and $\mathbf{MB}(X_i)$ contains exactly the variables X_j that are causes or effects of X_i (i.e., $\beta_{j,i} \neq 0$ or $\beta_{i,j} \neq 0$) and the spouse variables X_k (i.e., such that there exists a variable X_ℓ that is an effect of both X_i and X_k). A triplet (X_i, X_j, X_k) form a v-structure if X_i and X_j are causes of X_k while the first two are not directly linked.

3 Overview of DCILP

After describing the divide-and-conquer strategy at the core of DCILP, this section details the Integer Linear Programming approach used for the reconciliation of the local causal graphs. In the remainder of the paper, we assume the Markov property and causal sufficiency.

3.1 Divide-and-Conquer Strategy

As illustrated in Fig. 1, DCILP is a 3-phase process:

- Phase-1 aims to identify the Markov blankets $\mathbf{MB}(X_i)$ for each variable X_i with $i \in [d]$. Under the assumption that it is accurately identified, $\mathbf{MB}(X_i)$ contains all variables relevant to predict X_i (Tsamardinos et al. 2003), that is, the causes, effects and spouses of X_i.
- Phase-2 tackles the local causal subproblems defined on $\mathbf{S}_i := \{X_i\} \cup \mathbf{MB}(X_i)$, for $i \in [d]$. The restriction to the $\mathbf{S}_i$s usually makes the causal subproblems much smaller than the overall causal discovery problem; in counterpart, they no longer satisfy the causal sufficiency assumption as variables in $\mathbf{MB}(X_i)$ may be influenced by variables external to $\mathbf{MB}(X_i)$. This issue will be partially mitigated by only retaining the causal relations involving X_i, i.e. the causes and effects of X_i.
- Phase-3 tackles the building of a global causal graph based on the Markov blankets and the cause-effect edges respectively found in Phase-1 and Phase-2. This task is formalized as an ILP problem. ILP binary variables are associated with the cause-effect, spouse and v-structure relations among the causal variables. ILP constraints express the consistency of these relations. Lastly, the ILP objective states that the overall causal graph should be aligned to the best extent with the local causal graphs, subject to the consistency constraints.

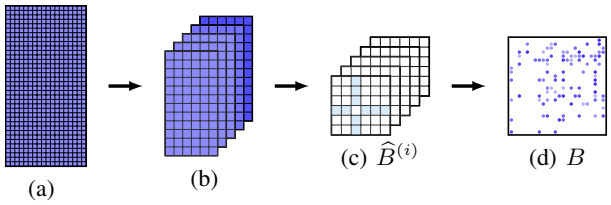


Figure 1: DCILP: (a) observational data; (b) data subsets on $\mathbf{S}_i$; (c) output matrix $\widehat{B}^{(i)}$ of Phase-2; (d) final solution B.

DCILP (Algorithm 1) defines a general framework, where different algorithms can be used in the three phases. Phase-1 extracts the $\mathbf{MB}(X_i)$s from the whole observational dataset $\mathcal{D}$. In Phase-2, the causal learning algorithm considers the local subproblems associated with the projection of $\mathcal{D}$ on the subset of variables $\mathbf{S}_i$s. This phase can be parallelized, estimating the causal subgraphs related with the $\mathbf{S}_i$ in an independent manner. Lastly, these relations are reconciled in Phase-3.

Algorithm 1 DCILP

Require: Observational data $\mathcal{X} \in \mathbb{R}^{n \times d}$

1: **(PHASE-1) Divide**:
   Estimate Markov blanket $\mathbf{MB}(X_i)$ for $i \in [d]$
2: **(PHASE-2) for** $i \in [d]$ **do in parallel**
3: $A^{(i)} \leftarrow$ Causal discovery on $\mathbf{S}_i := \{X_i\} \cup \mathbf{MB}(X_i)$
4: $\widehat{B}^{(i)}_{j,k} \leftarrow A^{(i)}_{j,k}$ if $j = i$ or $k = i$, and 0 otherwise
5: **(PHASE-3) Conquer:**
   $B \leftarrow$ Reconciliation from $\{\widehat{B}^{(i)}, i \in [d]\}$ via ILP

3.2 Conflicts Among Causal Subgraphs

The naive concatenation of the cause-effect relations found in Phase-2, given as:

$$\widehat{B} = \sum_{i=1}^{d} \widehat{B}^{(i)}. \tag{1}$$

generally fails to be a DAG, because $\widehat{B}^{(i)}$ might include undirected edges, and because some conflicts might exist among different $\widehat{B}^{(i)}$s.

Definition 3 (Conflicts among causal subgraphs)**.** Let $B^{(1)}$ and $B^{(2)}$ be defined as binary adjacency matrices among variables respectively ranging in V_1 and V_2, such that $V_1 \subset \mathbf{X}$ and $V_2 \subset \mathbf{X}$. There exists a conflict among $B^{(1)}$ and $B^{(2)}$ if and only if (iff) there exists a pair of variables (X_i, X_j) such that $\{X_i, X_j\} \subset V_1 \cap V_2$ with $B^{(1)}_{ij} \neq B^{(2)}_{ij}$.

Proposition 4. *For $i, j \in [d]$, let $\widehat{B}^{(i)}$ and $\widehat{B}^{(j)}$ respectively denote the i-th and j-th binary adjacency matrix output by Phase-2 (Algorithm 1, l. 4). A merge conflict between $\widehat{B}^{(i)}$ and $\widehat{B}^{(j)}$ is one of the following three types:*

1. (Type-1) *One of the two adjacency matrices includes a directed edge between X_i and X_j while the other includes no edge ($X_i \perp\!\!\!\perp X_j$).*
2. (Type-2) *The two adjacency matrices include edges between X_i and X_j with opposite directions: either one matrix contains an undirected edge between X_i and X_j while the other includes no edge; or both matrices includes a directed edge with opposite direction.*
3. (Type-3) *One of the two adjacency matrices include an undirected edge between X_i and X_j while the other includes a directed edge.*

The proof is given in (Dong et al. 2025, Appendix B).

As said, only the causal relations involving X_i are retained in $\widehat{B}^{(i)}$ (Algorithm 1). This restriction considerably reduces the set of conflicts compared to retaining all causal relations in $\widehat{A}^{(i)}$ (Algorithm 1, line 3).

When there exists a conflict between any pair of local solutions, and depending on its type, the naive concatenation $\widehat{B}$ (Eq. 1) is weighted accordingly; details in (Dong et al. 2025, Appendix C.2).

3.3 Reconciling Causal Subgraphs Through ILP

The reconciliation of the $\widehat{B}^{(i)}$s found in Phase-2 to form a consistent causal graph is formulated as an integer linear programming problem (Wolsey 2020). Let us denote B_{ij} the binary constraint variable representing the causal relation $X_i \to X_j$; let similarly S_{ij} denote the binary constraint variable representing the spouse relation between X_i and X_j. Finally, binary constraint variable V_{ijk} represent the v-structure $X_i \to X_k \leftarrow X_j$ among causal variables X_i, X_j and X_k:

$$\begin{aligned} B_{ij} &= 1 \text{ iff } X_i \to X_j \\ V_{ijk} = V_{jik} &= 1 \text{ iff there is a v-structure } (X_i \to X_k \leftarrow X_j) \\ S_{ij} = S_{ji} &= 1 \text{ iff } X_i \text{ and } X_j \text{ are spouses: } \exists k, V_{ijk} = 1. \end{aligned}$$

Let variables B_{ij}, S_{ij} and V_{ijk} be referred to as ILP variables for clarity in the following. The ILP variables are created if and only if they are consistent with the Markov blankets, that is, B_{ij} and S_{ij} are created if $X_i \in \mathbf{MB}(X_j)$ and $X_j \in \mathbf{MB}(X_i)$; and V_{ijk} is created if $X_i \in \mathbf{MB}(X_j) \cap \mathbf{MB}(X_k)$, $X_j \in \mathbf{MB}(X_i) \cap \mathbf{MB}(X_k)$, $X_k \in \mathbf{MB}(X_i) \cap \mathbf{MB}(X_j)$.

The following constraints defined on the ILP variables express the logical relations among Markov blankets and v-structures, for all i, j, k such that $i \neq j$, $j \neq k$, $k \neq i$:

$$B_{ij} + B_{ji} + S_{ij} \geq 1 \quad \text{if } \{i,j\} \subset (\mathbf{S}_i \cap \mathbf{S}_j) \tag{2}$$

$$V_{ijk} \leq B_{ik}, V_{ijk} \leq B_{jk} \quad \text{if } \{i,j,k\} \subset (\mathbf{S}_i \cap \mathbf{S}_j \cap \mathbf{S}_k) \tag{3}$$

$$B_{ik} + B_{jk} \leq 1 + V_{ijk} \quad \text{if } \{i,j,k\} \subset (\mathbf{S}_i \cap \mathbf{S}_j \cap \mathbf{S}_k) \tag{4}$$

$$V_{ijk} \leq S_{ij} \quad \text{if } \{i,j,k\} \subset (\mathbf{S}_i \cap \mathbf{S}_j \cap \mathbf{S}_k) \tag{5}$$

$$S_{ij} \leq \sum_k V_{ijk} \quad \text{if } \{i,j,k\} \subset (\mathbf{S}_i \cap \mathbf{S}_j \cap \mathbf{S}_k) \tag{6}$$

Proposition 5. *With same notations as above, under the assumption that Markov blanket $\mathbf{MB}(X_i)$ is true for all $i \in [d]$, the ILP variables associated with the sought causal graph B satisfy all linear constraints* (2)–(6).

Proof. Constraint (2) expresses that if X_j belongs to $\mathbf{MB}(X_i)$ it is either its cause, or its effect, or a spouse. The fact that $X_i \to X_k \leftarrow X_j$ is equivalent to constraints (3), (5), (4) results from the truth table of the involved ILP variables. Constraint (6) expresses that if X_i and X_k are spouses, then there exists at least one k such that i, j, k forms a v-structure $X_i \to X_k \leftarrow X_j$. □

The reconciliation of the causal subgraphs $\widehat{B}^{(i)}$ for $i \in [d]$ is formulated as the following ILP problem which searches (B, S, V) for B to be aligned with the naive concatenation $\widehat{B}$ of the causal subgraphs (Eq. 1) subject to the consistency constraints:

$$\max_{B,S,V} \langle \widehat{B}, B \rangle \text{ subject to constraints (2)–(6).} \tag{7}$$

3.4 Refining the ILP Formulation

For tractability, we further leverage the output $\widehat{B}^{(i)}$s of Phase-2, by imposing $B_{ij} = 0$ if no causal relations between X_i and X_j are found in $\widehat{B}^{(i)}$ or $\widehat{B}^{(j)}$:

$$B_{ij} = 0 \quad \text{if} \quad \widehat{B}^{(i)}_{ij} = \widehat{B}^{(i)}_{ji} = \widehat{B}^{(j)}_{ij} = \widehat{B}^{(j)}_{ji} = 0 \tag{8}$$

(i.e., if $\widehat{B}_{ij} = \widehat{B}_{ji} = 0$ in Eq. 1). The constraints (8) and (3) imply that, for all i, j, k mutually distinct:

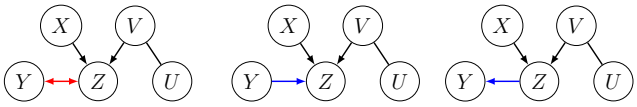
$$V_{ijk} = 0 \quad \text{if } \widehat{B}_{ik} = 0 \text{ or } \widehat{B}_{jk} = 0, \tag{9}$$

which further reduce the set of the ILP variables. Lastly, we add constraints to forbid 2-cycles:

$$B_{ij} + B_{ji} \leq 1 \quad \text{if } \{i, j\} \subset (\mathbf{S}_i \cap \mathbf{S}_j) \tag{10}$$

Therefore Phase-3 tackles the ILP problem (7) augmented with constraints (8), (9) and (10).

We note that the ILP resolution selects by design the edge orientations consistent with the Markov blankets (Fig. 2). DCILP thus recovers some properties of the PC algorithm (Spirtes et al. 2000), with the difference that it exploits the identification of the Markov blankets, as opposed to conditional independence tests.



(a) $\widehat{B}$ (b) B (c) B'

Figure 2: Consistency constraints based on Markov blankets information: if Y belongs to $\mathbf{MB}(V)$, the only feasible solution is B; otherwise it is B'.

The ILP resolution offers additional possibilities regarding the selection of the overall causal graph. In the implementation of DCILP, Gurobi's ILP solver (Gurobi Optimization 2025) is used for solving the ILP problem, and it enables returning a pool of ILP solutions attaining the same optimal value of the objective function. Then we select the solution with the best *DAGness value* based on (Zheng et al. 2018, Theorem 1): the DAGness of B is defined as $h(B) = \mathrm{tr}(\exp(B \odot B))$ satisfying $h(B) = d$ iff B is a DAG, and $h(B) > d$ otherwise. In the case where several ILP solutions have same h value, the sparsest one (with fewer edges) is selected.

3.5 Discussion

A main limitation of the presented framework is that errors in Phase-1 (e.g. variables missing or spuriously added in the Markov blankets) adversely affect Phase-2 and Phase-3. The impact of a missing variable X_j in $\mathbf{MB}(X_i)$ can be mitigated if X_i is present in $\mathbf{MB}(X_j)$;[2] otherwise, the edge cannot be recovered. On the contrary, the presence of spurious variables in $\mathbf{MB}(X_i)$ can to some extent be compensated for in Phase-2 and Phase-3. Another limitation is that the ILP solution is not guaranteed to be a DAG; constraints (10) only rule out 2-cycles. The ILP formulation however makes it possible to incrementally add new constraints, to rule out the cycles present in former solutions. The question is whether the incremental addition of such constraints yield a DAG solution in a (sufficiently) small number of iterations. These limitations can be addressed by choosing appropriate algorithms involved in Phase-1 or Phase-2. For instance, Phase-1 currently exploits the precision matrix of $\mathbf{X}$ to identify the Markov blankets under the Gaussian linear SEM assumption. An alternative in non-linear cases is to leverage feature selection algorithms (Tsamardinos et al. 2003) to identify the smallest subset of variables needed to predict a given variable X_i.

Along the same line, the choice of the causal learning algorithm in Phase-2 may depend on the prior knowledge about the data at hand, e.g., whether the causal variables are expected to satisfy the equal-variance assumption (EV case) or not (NV case). Some algorithms such as GES are more robust to the so-called varsortability bias (Reisach, Seiler, and Weichwald 2021) than others (e.g., NOTEARS and DAGMA (Bello, Aragam, and Ravikumar 2022)).

Complexity. In Phase-1, the identification of the Markov blankets is conducted under the assumption of Gaussian graphical models. Letting $\Sigma = \mathrm{cov}(\mathbf{X})$ denote the covariance matrix of the data, and noting $\Theta = \Sigma^{-1}$ its inverse (the precision matrix), it follows from the Hammersley–Clifford theorem that the support of $\Theta_{i,:}$ is exactly $\mathbf{S}_i = \mathbf{MB}(X_i) \cup \{X_i\}$. Its complexity is bounded by $O(d^3)$, corresponding to the inversion of the symmetric positive definite matrix Σ. More advanced methods with reduced complexity can be used; see, e.g., (Hsieh et al. 2014).

In Phase-2, the causal subproblems, restricted to the subset of variables $\mathbf{S}_i$, are solved independently for $i \in [d]$. Letting d_{MB} denote the maximal size of the estimated Markov blankets, the computational complexity of Phase-2, considering that the subproblems are handled in parallel and using e.g. DAGMA (Bello, Aragam, and Ravikumar 2022), thus is $O(d_{MB}^3)$.

In Phase-3, the number of ILP variables B_{ij} and S_{ij} is bounded by $O(d_{MB}d)$ with usually $d_{MB} \ll d$. The number

[2]Phase-1 currently determines the Markov blankets from the precision matrix of $\mathbf{X}$, which is symmetric; see (Dong et al. 2025, Appendix C.1).

of V_{ijk} variables is bounded by $O(d^2_{MB}d)$ in view of (9). The ILP problem is NP-hard and becomes intractable when the number of variables and the size of the Markov blankets increases beyond a certain range. The observation from experimental results is that the computational gains in Phase-2 (solving in parallel the d subproblems as opposed to learning one causal graph of d nodes) cancel out the computational cost of Phase-3, for d ranging from medium to large-sized.

4 Experiments

This section reports on the experimental validation of DCILP, referring to the extended version (Dong et al. 2025) for more details and complementary results.

4.1 Experimental Setting

Goals. The primary goal of the experiments is to evaluate the performance of DCILP according to the standard SHD, TPR, FDR and FPR indicators for causal learning, together with its computational efficiency.

A second goal is to assess how the causal learner used in DCILP' Phase-2 influences the overall performances. We report on the performances of DCILP-ges (respectively DCILP-dagma), corresponding to DCILP using GES (resp. DAGMA) during Phase-2. The choice of GES[3] (Chickering 2002b) and DAGMA (Bello, Aragam, and Ravikumar 2022), known as a notable refinement of NOTEARS (Zheng et al. 2018), is used as they are two representative state-of-the-art causal learning methods using different techniques: GES is a greedy search method (optimal in the large sample limit) for finding the CPDAG while DAGMA is an efficient continuous optimization method for learning causal DAGs. DCILP-ges and DCILP-dagma are assessed against the GES and DAGMA baselines; GOLEM (Ng, Ghassami, and Zhang 2020) and DAS (Montagna et al. 2023) are also used for comparison.[4] We also examine the impact of the Phase-1 performance on the overall result, by considering the DCILP (MB*) variants where the ground truth Markov blankets are supplied to Phase-2.

Benchmarks. Following (Zheng et al. 2018), we consider synthetic and real-world datasets. The synthetic observational datasets are generated from linear SEMs, where the causal graph $B^\star$ is drawn from random DAGs with an Erdős–Rényi (ER) or scale-free (SF) model. The number d of variables ranges in $\{50, 100, 200, 400, 800, 1000, 1600\}$. The number n of samples is set to $5\times, 10\times$ or $50 \times d$. The real-world dataset is generated using the so-called MUNIN model (Gu and Zhou 2020; Andreassen et al. 1989), which is a Bayesian network with $d = 1{,}041$ nodes that models a medical expert system based on electromyographs (EMG) to assist diagnosis of neuromuscular disorders. Datasets are generated according to the considered graph $B^\star$, with coefficients (edge weights) drawn from the uniform distribution $\text{Unif}([-2, -0.5] \cup [0.5, 2])$. The noise variables of the SEM data are from either Gaussian, Gumbel or uniform distribution. All reported results are median values from 3 to 5 runs with different random seeds.

Computational environment. The parallelization of DCILP' Phase-2 is conducted on a cluster via the SLURM scheduler; each experiment uses at most 400 CPU cores. For $d \leq 200$, Phase-2 is distributed among $2d$ CPU cores; each local causal problem is handled on 2 cores. For higher values of d, the parallelization of Phase-2 is then limited by congestion. The baseline algorithms and DCILP' Phase-1 run on one CPU core, and DCILP' Phase-3 runs on four CPU cores; CPU specifications are detailed in (Dong et al. 2025, Appendix D.2).

4.2 Results on Synthetic Graphs

DCILP-ges. Figure 3 and Table 2 illustrate the performances of DCILP-ges and DCILP-ges (MB*) on the ER2 data with $n = 50d$ and on ER1 data with $n = 10d$ respectively, in comparison with GES and the other baselines.

On these problems, DCILP outperforms GES, particularly in terms of FDR and computational time, except for small problem dimensions ($d \leq 200$), as GES is very fast on smaller problems. DCILP-ges and DCILP-ges (MB*) yield identical results, as might be expected given the favorable n/d ratio. The computational time for DCILP-ges increases with d, though at a much slower rate than for GES. The distribution of computational time among the DCILP phases shows that the majority of the time is spent in Phase-2, increasing with d.

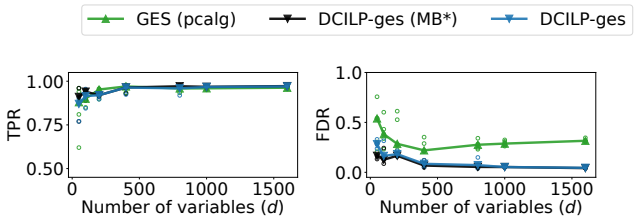


Figure 3: Learning accuracy of GES, DCILP-ges, and DCILP-ges (MB*) on ER2 data.

DCILP-ges achieves significant gains over GES in terms of DAG learning accuracy, which can be explained from two facts. Firstly, the causal learning subproblems in DCILP' Phase-2 enjoy a higher *effective sample size*, in the sense that the ratio $n/|\mathbf{MB}(X_i)|$ is higher than for the global problem. Secondly, the reconciliation achieved by Phase-3 among the causal subgraphs tends to remove spurious directions and it avoids 2-cycles, while GES focuses on producing a CPDAG and might thus retain undirected edges.

Fig. 4 suggests that the speed-up achieved by DCILP-ges over GES increases for $d \geq 400$. For $d = 1600$, DCILP-ges achieves more than $5\times$ speedup over GES on ER2 data.

DCILP-dagma. Table 1 and Table 2 show the performances of DCILP-dagma and DCILP-dagma (MB*) on SF3 data (for $n/d = 20$) and ER1 data (for $n/d = 10$) respectively, in comparison with DAGMA and the other baselines.

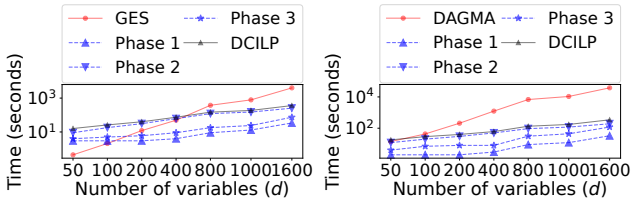
[3]We used the implementation from the R package `pcalg` (https://cran.r-project.org/web/packages/pcalg/index.html) , which is a most efficient version of GES to our knowledge.

[4]Results of DAS are reported from (Montagna et al. 2023).

d	Algorithm	TPR	FDR	SHD	# of edges	time (sec)
400	GOLEM	0.987	0.025	46 ± 14.47	1208	4437.526
	GES	0.907	0.471	1029 ± 237.57	2048	1330.314
	DAGMA	0.999	0.003	3 ± 1.53	1196	3997.719
	DCILP-dagma (MB*)	0.964	0.135	208 ± 24.43	1330	1102.636
	DCILP-dagma	0.925	0.177	322 ± 68.83	1336	974.538
800	GOLEM	0.964	0.049	187 ± 355.73	2426	25533.171
	GES	0.878	0.518	2444 ± 1182.74	4068	12063.320
	DAGMA	0.999	0.003	7 ± 8.66	2398	22952.589
	DCILP-dagma (MB*)	0.938	0.250	879 ± 234.64	2994	4104.407
	DCILP-dagma	0.923	0.250	914 ± 191.78	2932	3405.801

Table 1: SF3 ($d \in \{400, 800\}, n/d = 20$): Comparative performance of GOLEM, DAGMA, GES, and DCILP-dagma. DCILP-dagma uses 400 CPU cores in Phase-2.

We observe that DAGMA almost exactly recovers the underlying DAGs, with TPR and FDR respectively close to 1 and 0. DCILP-dagma achieves significant gains over DAGMA in running time for all problem dimensions d, with a moderate loss in learning accuracy (median TPR circa 0.9; median FDR around 0.2 on SF3 data and under 0.1 on ER1 data). Complementary results in the non-equal noise variance (NV) setting (Reisach, Seiler, and Weichwald 2021) show however that DCILP-ges is much more robust in the NV case than DAGMA (Figure 5). More details are presented in the extended version (Dong et al. 2025).



(a) DCILP-ges (b) DCILP-dagma

Figure 4: Computation time of the 3 phases on ER2 data with different graph sizes (d).

d	Algorithm	SHD	#edges	time (sec)
200	GOLEM	6.0 ± 0.58	194	686.28
	DAS	110 ± 6.60	–	282.00
	GES	177.2 ± 10.21	361	4.58
	DCILP-ges	53 ± 16.17	243	68.33
	DAGMA	1.6 ± 1.67	200	173.93
	DCILP-dagma	15 ± 5.57	208	59.00
1000	GOLEM	31.0 ± 8.19	990	7885.99
	DAS	994 ± 15.00	–	5544.00
	GES	1521.8 ± 51.3	2470	629.30
	DCILP-ges	79 ± 5.86	1004	328.00
	DAGMA	3.40 ± 2.88	1001	8554.24
	DCILP-dagma	46 ± 23.59	1014	272.67

Table 2: Results on ER1 data ($n/d = 10$): Comparative performance of DAS, DAGMA, GES, DCILP-ges and DCILP-dagma. DCILP-dagma and DCILP-ges use 400 CPU cores in Phase-2.

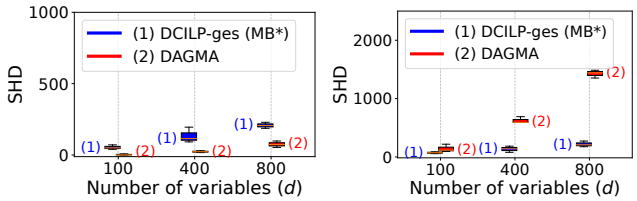


Figure 5: Results on ER2 data under the EV (left) and NV (right) settings.

In all settings, the time efficiency gains of DCILP-dagma over DAGMA are even more significant than for DCILP-ges compared to GES: DCILP-dagma achieves an $100\times$ speed up over DAGMA for $d \geq 400$. As discussed in Section 3.5, such speedups are largely due to the parallelization of the local learning subproblems in Phase-2.[5]

4.3 Results on Real-World Graph

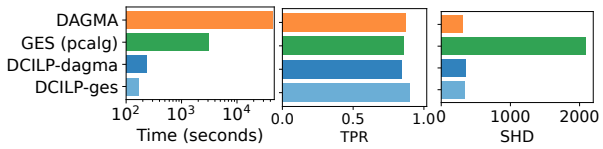
On MUNIN (Fig. 6), DCILP-dagma and DCILP-ges achieve significant reductions in running time compared to DAGMA (with a speed-up of circa 270 times) and compared to GES (with a speed-up of circa 25 times). At the same time, their learning accuracy is on par with that of DAGMA, and considerably better than for GES; the SHDs of DCILP-dagma and DCILP-ges rank first or second along with DAGMA; the TPRs are similar for DAGMA, GES and DCILP-dagma, and slightly inferior for DCILP-ges with uniform noise.

Similar trends are observed for other n/d ratios and noise types (details in (Dong et al. 2025, Appendix D.6)).

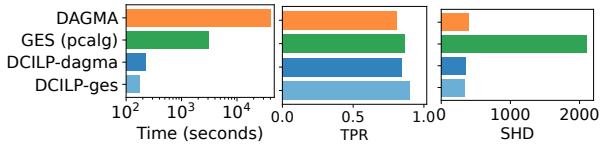
4.4 Impact of ILP in Phase-3

The actual effects of the ILP reconciliation process are analyzed by comparing the performance of DCILP-dagma and DCILP-ges with that of the naive merge $\widehat{B}$ (Eq. (1)) of the causal graphs obtained in Phase-2, on the ER2 and SF3 cases. As shown in Fig. 7, the main impact of the ILP-based reconciliation is that it significantly reduces FDR, at the expense of a slight reduction in TPR, particularly so for high values of d (TPR remaining above 90%).

[5]Despite the fact that the parallelization of Phase-2 encounters congestion for large d given the available CPU cores.



(a) Gaussian noise



(b) Uniform noise

Figure 6: Comparative results on MUNIN linear SEM with Gaussian (a) and Uniform (b) noise distributions. The ratio n/d is 5.

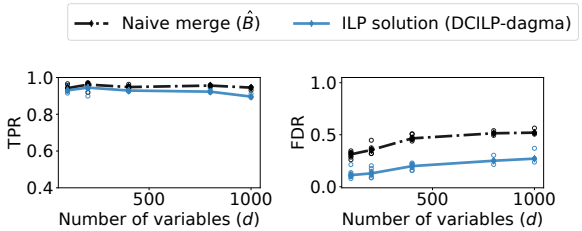


Figure 7: Comparison with the naive merge $\widehat{B}$: DCILP-dagma on SF3 data.

5 Position with Respect to Related Work

The most related work is the divide-and-conquer strategy proposed by (Gu and Zhou 2020), defining the *Partition, Estimation and Fusion* (PEF) approach. The partition step proceeds as a hierarchical clustering algorithm. The estimation step consists of estimating the DAG or CPDAG associated with each cluster. A first difference between PEF and DCILP regards the divide phase: in PEF, variables are partitioned while DCILP considers overlapping Markov blankets. Noting that both approaches face the causal insufficiency issue, the redundancy involved in DCILP-Phase-2 possibly results in a better robustness. The second and most important difference between PEF and DCILP regards the conquer phase. PEF tackles a combinatorial optimization problem, defining and selecting the best edges among the subgraphs, while DCILP relies on the ILP resolution to ensure the consistency of the global causal graph built from the local ones.

A top-down strategy, proposed by (Zhang et al. 2020) and aimed at CPDAG, proceeds by recursively splitting the set of variables in smaller subsets. Inversely, a bottom-up strategy is the graph growing structure learning (GGSL) proposed by (Gao, Fadnis, and Campbell 2017) and aimed to Bayesian Networks. In these approaches, a main goal is to reduce the number of conditional independence tests needed to elicit the overall causal structure. Another related work, though to a lesser extent, is the recursive variable elimination method named MARVEL proposed by (Mokhtarian et al. 2021). Like DCILP, the MARVEL algorithm relies on the identification of Markov blankets, using e.g. Grow-Shrink (GS) (Margaritis and Thrun 1999), IAMB (Tsamardinos et al. 2003), or precision matrix-based methods (e.g., (Loh and Bühlmann 2014)). The main issue in practice is that the recursive process offers no room to recover from potential early mistakes.

(Jaakkola et al. 2010) tackle the problem of learning Bayesian network structures, where the acyclicity constraint is handled using linear programming relaxations. (Cussens et al. 2017; Cussens 2023) proposed integer linear programming methods for Bayesian network learning, by which the IP problem considers binary variables $x_{i \leftarrow J}$ defined on the ensemble of subsets as indicators of whether a subset J is the parent set of X_i. This problem setting is proposed for exact learning of Bayesian networks (DAGs) based on a decomposable score function; in particular, it does not make use of Markov blankets, and involves much more decision variables than DCILP.

6 Conclusion

The divide-and-conquer DCILP framework is based on the natural decomposition of the desired causal graph defined by the Markov blankets $\mathbf{MB}(X_i)$ associated with each variable X_i. The main contribution of the paper is to demonstrate that the causal relationships involving X_i – learned in parallel by focusing on $\mathbf{MB}(X_i)$ – can be reconciled using an ILP formulation, and that this reconciliation can be efficiently handled by an ILP solver. This three-phase framework accommodates a variety of choices for each phase, from identifying the Markov blankets in Phase-1 to causal learning in Phase-2 and selecting the ILP solver in Phase-3. The experimental results effectively demonstrate the framework's merits, particularly so on the large-scale real-world MUNIN data, achieving a computational time reduced by two orders of magnitude with on-par SHD compared to DAGMA (respectively, one order of magnitude in computational time with a significant gain in SHD compared to GES).

A main limitation of the approach comes from the learning accuracy of the Markov blankets and the local causal relations in Phase-1 and Phase-2, which require a sufficient samples-to-variables ratio.

One avenue for future research is to consider the three phases in a more integrated manner, e.g., taking into account the strength of the causal edges identified in Phase-2 within the ILP objective in Phase-3 (as opposed to only taking into account the existence of such edges).

Another perspective is to leverage the multiple solutions discovered by the ILP solver in Phase-3. For instance, the causal relations retained in several ILP solutions can be used to define the backbone of the desired causal graph, thus extending DCILP in the spirit of ensemble learning. A longer-term perspective is to approximate well-studied causal scores (Loh and Bühlmann 2014) using bilinear functions, in order to extend Phase-3, from integer linear programming to quadratic programming.

Acknowledgments

We thank the anonymous reviewers for their helpful comments and suggestions. The first author was funded by a grant from Fujitsu, and the work was also supported by the CAUSALI-T-AI project of PEPR-IA (ANR-23-PEIA-0007 CAUSALI-T-AI).

References

Andreassen, S.; Jensen, F.; Andersen, S.; Falck, B.; Kjærulff, U.; Woldbye, M.; Sørensen, A.; Rosenfalck, A.; and Jensen, F. 1989. *MUNIN: an expert EMG assistant*, 255–277. Pergamon Press. ISBN 0444811060.

Aragam, B.; Amini, A.; and Zhou, Q. 2019. Globally optimal score-based learning of directed acyclic graphs in high-dimensions. *Advances in Neural Information Processing Systems*, 32.

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Bello, K.; Aragam, B.; and Ravikumar, P. 2022. DAGMA: Learning dags via m-matrices and a log-determinant acyclicity characterization. *Advances in Neural Information Processing Systems*, 35: 8226–8239.

Chickering, D. M. 2002a. Learning equivalence classes of Bayesian-network structures. *The Journal of Machine Learning Research*, 2: 445–498.

Chickering, D. M. 2002b. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov): 507–554.

Cussens, J. 2023. Branch-Price-and-Cut for Causal Discovery. In *2nd Conference on Causal Learning and Reasoning*.

Cussens, J.; Järvisalo, M.; Korhonen, J. H.; and Bartlett, M. 2017. Bayesian network structure learning with integer programming: Polytopes, facets and complexity. *Journal of Artificial Intelligence Research*, 58: 185–229.

Dong, S.; Sebag, M.; Uemura, K.; Fujii, A.; Chang, S.; Koyanagi, Y.; and Maruhashi, K. 2025. DCILP: A distributed approach for large-scale causal structure learning. *arXiv preprint arXiv:2406.10481*.

Gao, T.; Fadnis, K.; and Campbell, M. 2017. Local-to-global Bayesian network structure learning. In *International Conference on Machine Learning*, 1193–1202. PMLR.

Gu, J.; and Zhou, Q. 2020. Learning big Gaussian Bayesian networks: Partition, estimation and fusion. *Journal of machine learning research*, 21(158): 1–31.

Gurobi Optimization. 2025. Gurobi Optimizer Reference Manual. https://docs.gurobi.com/current/. Accessed: 2025-02-10.

Hsieh, C.-J.; Sustik, M. A.; Dhillon, I. S.; and Ravikumar, P. 2014. QUIC: Quadratic Approximation for Sparse Inverse Covariance Estimation. *Journal of Machine Learning Research*, 15(83): 2911–2947.

Jaakkola, T.; Sontag, D.; Globerson, A.; and Meila, M. 2010. Learning Bayesian network structure using LP relaxations. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 358–365. JMLR Workshop and Conference Proceedings.

Loh, P.-L.; and Bühlmann, P. 2014. High-dimensional learning of linear causal networks via inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1): 3065–3105.

Lopez, R.; Hütter, J.-C.; Pritchard, J.; and Regev, A. 2022. Large-scale differentiable causal discovery of factor graphs. *Advances in Neural Information Processing Systems*, 35: 19290–19303.

Margaritis, D.; and Thrun, S. 1999. Bayesian network induction via local neighborhoods. *Advances in neural information processing systems*, 12.

Meek, C. 1995. Causal Inference and Causal Explanation with Background Knowledge. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, 403–410. Cambridge, MA: Morgan Kaufmann Publishers Inc. ISBN 1558603859.

Mokhtarian, E.; Akbari, S.; Ghassami, A.; and Kiyavash, N. 2021. A recursive Markov boundary-based approach to causal structure learning. In *The KDD'21 Workshop on Causal Discovery*, 26–54. PMLR.

Montagna, F.; Noceti, N.; Rosasco, L.; Zhang, K.; and Locatello, F. 2023. Scalable causal discovery with score matching. In *Conference on Causal Learning and Reasoning*, 752–771. PMLR.

Ng, I.; Ghassami, A.; and Zhang, K. 2020. On the role of sparsity and DAG constraints for learning linear DAGs. *Advances in Neural Information Processing Systems*, 33: 17943–17954.

Ng, I.; Zheng, Y.; Zhang, J.; and Zhang, K. 2021. Reliable Causal Discovery with Improved Exact Search and Weaker Assumptions. *Advances in Neural Information Processing Systems*, 34: 20308–20320.

Pearl, J. 2000. *Causality: Models, Reasoning and Inference*. Cambridge University Press.

Peters, J.; Bühlmann, P.; and Meinshausen, N. 2016. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 947–1012.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Ramsey, J.; Glymour, M.; Sanchez-Romero, R.; and Glymour, C. 2017. A million variables and more: the fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International journal of data science and analytics*, 3: 121–129.

Reisach, A.; Seiler, C.; and Weichwald, S. 2021. Beware of the simulated DAG! Causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing Systems*, 34: 27772–27784.

Sauer, A.; and Geiger, A. 2021. Counterfactual Generative Networks. In *International Conference on Learning Representations (ICLR)*.

Spirtes, P.; Glymour, C. N.; Scheines, R.; and Heckerman, D. 2000. *Causation, prediction, and search*. Cambridge, MA: MIT Press.

Tsamardinos, I.; Aliferis, C. F.; Statnikov, A. R.; and Statnikov, E. 2003. Algorithms for large scale Markov blanket discovery. In *FLAIRS conference*, volume 2, 376–380. St. Augustine, FL.

Tsirtsis, S.; Karimi, A.-H.; Lucic, A.; Gomez-Rodriguez, M.; Valera, I.; and Lakkaraju, H. 2021. ICML Workshop on Algorithmic Recourse.

Wolsey, L. A. 2020. *Integer programming*. Hoboken, NJ: John Wiley & Sons.

Wu, X.; Jiang, B.; Wu, T.; and Chen, H. 2023. Practical Markov boundary learning without strong assumptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 10388–10398.

Wu, X.; Jiang, B.; Yu, K.; Miao, c.; and Chen, H. 2020. Accurate Markov Boundary Discovery for Causal Feature Selection. *IEEE Transactions on Cybernetics*, 50(12): 4983–4996.

Wu, X.; Jiang, B.; Zhong, Y.; and Chen, H. 2022. Multi-Target Markov Boundary Discovery: Theory, Algorithm, and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, H.; Zhou, S.; Yan, C.; Guan, J.; Wang, X.; Zhang, J.; and Huan, J. 2020. Learning causal structures based on divide and conquer. *IEEE Transactions on Cybernetics*, 52(5): 3232–3243.

Zheng, X.; Aragam, B.; Ravikumar, P. K.; and Xing, E. P. 2018. DAGs with NO TEARS: Continuous Optimization for Structure Learning. In *Advances in Neural Information Processing Systems*, volume 31.