Proceedings of Machine Learning Research 343:182–209, 2026 Probabilistic Graphical Models (PGM)

Learning Causal Structure of Time Series using Best Order Score Search

Irene Gema Castillo Mansilla IRENE.CASTILLO@UNI-POTSDAM.DE
University of Potsdam
Potsdam, Germany

Urmi Ninad URMI.NINAD@UNI-POTSDAM.DE
Department of Computer Science
University of Potsdam
Potsdam, Germany

Editors: Gustau Camps-Valls, Manuele Leonelli and Gherardo Varando

Abstract

Causal structure learning from observational data is central to many scientific and policy domains, but the time series setting common to many disciplines poses several challenges due to temporal dependence. In this paper we focus on score-based causal discovery for multivariate time series and introduce TS-BOSS, a time series extension of the recently proposed Best Order Score Search (BOSS) (Andrews et al., 2023). TS-BOSS performs a permutation-based search over dynamic Bayesian network structures while leveraging grow–shrink trees to cache intermediate score computations, preserving the scalability and strong empirical performance of BOSS in the static setting. We provide theoretical guarantees establishing the soundness of TS-BOSS under suitable assumptions, and we present an intermediate result that extends classical subgraph minimality results for permutation-based methods to the dynamic (time series) setting. This result paves the way for time series extensions of sparsest permutation-based causal discovery algorithms under suitable stationarity assumptions. We present experimental results on synthetic data that highlight the high accuracy of TS-BOSS.

Keywords: Causal discovery, score-based methods, time series, BIC score

1. Introduction

Learning the causal structure underlying observational data is fundamental across a wide range of disciplines, from policy-relevant fields such as economics and epidemiology to natural sciences including neuroscience and Earth system science. Causal structure learning methods can be broadly categorized into constraint-based and score-based methods. The former make use of conditional independence constraints learned from the data to infer causal relationships, while the latter optimize a consistently-defined score defined over the space of directed acyclic graphs (DAGs). Further methods exist such as restricted structural causal model (SCM) framework which employs assumptions on functional relationships and noise distributions to uncover causal relationships. See (Pearl, 2009; Spirtes et al., 2000; Peters et al., 2017) for details on the different approaches and (Glymour et al., 2019; Squires and Uhler, 2023) for a review.

Often, the observational data take the form of time series, with observations at each time point influenced by past observations. The challenges specific to time series causal

© 2026 I.G. Castillo Mansilla & U. Ninad.

structure learning, or time series *causal discovery*, have been discussed at length (Runge et al., 2019; Camps-Valls et al., 2023). For instance, in the setting of a single multivariate time series, temporal dependence violates the i.i.d. assumption, thereby limiting the theoretical guarantees of causal discovery methods and statistical procedures more generally. Several methods spanning the different approaches for causal discovery have been proposed for the time series setting (Hyvärinen et al., 2010; Runge, 2020; Pamfil et al., 2020), see (Assaad et al., 2022) for a survey.

In this work, our focus will be the score-based approach to causal discovery, while presenting experimental comparisons to established algorithms for time series causal discovery. Recently, *best order score search* (BOSS), a score-based method for structure learning, was introduced (Andrews et al., 2023). BOSS performs a permutation-based search over DAGs and leverages a novel data structure—grow–shrink trees (GSTs)—to cache intermediate computations. BOSS was shown to achieve high accuracy and state-of-the-art performance while maintaining scalability to a large number of nodes. We propose a time series extension of BOSS for causal discovery over multivariate time series data named **TS-BOSS**. We provide theoretical results for the soundness of TS-BOSS and ablation studies that compare TS-BOSS to standard algorithms for time series causal discovery. Our results show that TS-BOSS achieves a better overall precision–recall trade-off than the competing methods, especially in the high autocorrelation regime where high sample dependence would be expected to hamper performance.

This establishes TS-BOSS as a high-performing and scalable time series causal discovery method. We also provide an intermediate result (Theorem 5) that extends the result of (Verma and Pearl, 1990) (Theorem 2 and Corollary 2) to the time series (i.e. dynamic bayesian network) setting. This result has the useful implication that sparsest permutation-based methods for causal discovery (Raskutti and Uhler, 2018) can be extended to the time series setting under suitable stationarity assumptions.

The remainder of this paper is organized as follows: In Section 2 we present the related work, and in Section 3 we provide the general background for the time series causal discovery problem. In Section 4 we present our algorithm TS-BOSS and in Section 5 we provide theoretical results that guarantee soundness of TS-BOSS. In Section 6 we present a simulation study on synthetic data and conclude with a discussion in Section 7.

2. Related work

Greedy Equivalence Search (GES) was an influential early method in score-based causal discovery which leveraged Meek's conjecture to search greedily over equivalence classes of DAGs (Chickering, 2003). Causal discovery was presented as a search problem over permutations that yield the sparsest DAG in (Raskutti and Uhler, 2018) with the *sparsest permutation* (SP) algorithm. Solus et al. (2021); Lam et al. (2022) proposed greedy score-based approaches for the permutation search over DAGs and Andrews et al. (2023) proposed an efficient algorithm within the same class of methods using Grow-Shrink trees. NOTEARS algorithm of Zheng et al. (2018) presented the search over space of DAGs as a continuous optimization problem, and Pamfil et al. (2020) extended this work to the dynamic setting with DYNOTEARS. Several other continuous optimization-based methods and perspectives thereon have been proposed since then (Bello et al., 2022; Reisach et al., 2021; Ng et al.,

2024). Recently, Wienöbst et al. (2025) proposed a discrete-search based method using a modified grow-shrink procedure that provides significant speed-up as graph size increases.

In the dynamic setting, Runge (2020) proposed a constraint-based causal discovery algorithm for time series using a modified conditional independence (CI) test to mitigate the effects of ill-calibrated CI tests in the single multivariate time series setting while allowing for contemporaneous edges. In (Malinsky and Spirtes, 2018), a hybrid constraint and score-based method for time series causal discovery with latent confounders was introduced, that uses GES as a model selection step.

3. Time Series Causal Discovery

3.1. Preliminaries and Problem Setup

Let $\mathbf{X}_t = (X_t^1, \dots, X_t^m)$ denote a vector of m random variables observed at discrete time points $t \in \mathbb{Z}$. The collection $(\mathbf{X}_t)_{t\in\mathbb{Z}}$ is called a *multivariate time series*. We assume that dynamical processes that are representable as multivariate time series can be modeled as a *time series SCM* (or ts-SCM):

$$X_t^j := f^j(\mathrm{Pa}(X_t^j), \eta_t^j) \qquad \forall\ X_t^j \in \mathbf{X}_t \ \text{ and } \ \forall\ t \in \mathbb{Z}\ , \tag{1}$$

with ':=' denoting structural assignments. Here, each f^j denotes the causal mechanism corresponding to X_t^j that models each X_t^j in terms of the parents of X_t^j, represented by $\mathrm{Pa}(X_t^j)$, and the random variable η_t^j which is the noise associated to X_t^j. For each $t \in \mathbb{Z}$, the noises $\boldsymbol{\eta}_t = (\eta_t^1, \dots, \eta_t^m)$ are assumed to be jointly independent. Furthermore, for each $i \in [m]$, the noises $\boldsymbol{\eta}^i = (\eta_t^i)_{t\in\mathbb{Z}}$ are assumed to be i.i.d.. The time series SCM (1) entails a *time series graph* with nodes X_t^i for all $i \in [m]$ and for all $t \in \mathbb{Z}$, and directed edges from a node X_t^i to a node $X_{t'}^j$ if and only if $X_t^i \in \mathrm{Pa}(X_{t'}^j)$, for all $i, j \in [m]$ and $t, t' \in \mathbb{Z}$ such that $t \leq t', (i,t) \neq (j,t')$. The latter condition excludes self-cycles in the time series graph, and we additionally exclude general directed cycles. The resultant graph is referred to as a *time series directed acyclic graph* or a *ts-DAG* (Gerhardus, 2024). The concept of d-separation (Pearl, 2009) of two disjoint sets of vertices given a third extends naturally to ts-DAGs if the sets are finite. Given finite disjoint subsets $\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3$ of $(\mathbf{X}_t)_{t\in\mathbb{Z}}$, we denote the statement $\mathbf{S}_1$ is d-separated of $\mathbf{S}_2$ given $\mathbf{S}_3$ in a ts-DAG $\mathcal{G}_{ts}$ as $\mathbf{S}_1 \perp\!\!\!\perp_d \mathbf{S}_2 \mid \mathbf{S}_3$. DAGs with the same set of d-separations are represented by a *completed partially directed acyclic graph* (CPDAG), and a CPDAG with additional edge orientations due to background knowledge, such as time order, is referred to as the *maximally oriented partially directed acyclic graph* (MPDAG). Accordingly, we define the time series versions of these graphical objects namely ts-CPDAG and ts-MPDAG, respectively.

In this work, the focus is causal discovery of the ts-DAG $\mathcal{G}_{ts}$, entailed by SCM (1) underlying the multivariate time series $(\mathbf{X}_t)_{t\in\mathbb{Z}}$. We make the following assumptions relating graphical d-separation in $\mathcal{G}_{ts}$ and (conditional) independence statements of a finite collection of random variables X_t^i to enable causal discovery. Following the notation of Hochsprung et al. (2024), we denote finite disjoint subsets $I_1, I_2, I_3 \subset [m] \times \mathbb{Z}$ and corresponding finite disjoint subsets $\mathbf{S}_m = \{X_t^i \ : \ (i,t) \in I_m\}$ for $m = \{1,2,3\}$ for the following assumptions.

Assumption 1 (Time series local Markov property) *For every $X_t^i \in (\mathbf{X}_t)_{t\in\mathbb{Z}}$ and every finite set $\mathbf{S}_i \subset (\mathbf{X}_t)_{t\in\mathbb{Z}}$, such that $\mathbf{S}_i$ contains no descendant or parent of X_t^i, $X_t^i \perp\!\!\!\perp \mathbf{S}_i | Pa(X_t^i)$, where parents and descendants are defined w.r.t. the ts-DAG $\mathcal{G}_{ts}$.*

Assumption 2 (Time series faithfulness property) *For any finite disjoint sets I_1, I_2 and I_3, with $I_j \subseteq [m] \times \mathbb{Z}$ for $j = 1, 2, 3$, if $\mathbf{S}_1 \perp\!\!\!\perp \mathbf{S}_2 \mid \mathbf{S}_3$, then $\mathbf{S}_1 \perp\!\!\!\perp_d \mathbf{S}_2 \mid \mathbf{S}_3$ in $\mathcal{G}_{ts}$.*

Furthermore, the requirement that noises $\boldsymbol{\eta}_t = (\eta_t^1, \ldots, \eta_t^m)$ are jointly independent imposes *causal sufficiency*, namely there are no latent confounders. For two nodes X_s^i, X_t^j in $\mathcal{G}_{ts}$ such that $s \leq t$ and $X_s^i \in \mathrm{Pa}(X_t^j)$, the difference $\tau = t - s$ is referred to as the *time lag* corresponding to the edge $X_s^i \to X_t^j$. In accordance with the time-series SCM formulation, causal relationships are restricted to respect the temporal order of the data, such that variables at future time points cannot be parents of variables at earlier time points. Note that we allow for zero time lags (i.e. contemporaneous edges), in order to account for insufficient time resolution of the time series.

Assumption 3 (Maximum time lag) *The time-series causal graph $\mathcal{G}_{ts}$ is assumed to have a known finite maximum time lag $\tau_{\max} \geq 0$, such that every edge in $\mathcal{G}_{ts}$ has an associated time lag $\tau \leq \tau_{\max}$.*

Assumption 4 (Stationary causal structure) *The causal structure of ts-SCM (1) is* stationary*, namely the causal mechanisms f^i corresponding to X_t^i in Equation (1) remain constant for all $t \in \mathbb{Z}$.*

Assumption 4 implies that the ts-DAG $\mathcal{G}_{ts}$ entailed by ts-SCM (1) has the *repeating edge property* ($X_{t'-s'}^j \to X_{t'}^i \Leftrightarrow X_{t-s'}^j \to X_t^i$) because of which the edges with an arrowhead incident on $\mathbf{X}_t$ for any $t \in \mathbb{Z}$ are sufficient to reconstruct the entire causal graph $\mathcal{G}_{ts}$ (Gerhardus, 2024). $\mathbf{X}_t$, for an arbitrary t, will be referred to as the *contemporaneous slice* of time series $(\mathbf{X}_t)_{t \in \mathbb{Z}}$. By Assumption 3 and 4, we can conclude that the following target object of time series causal discovery yields $\mathcal{G}_{ts}$ (Assaad et al., 2022).

Definition 1 (Window causal graph) *Let $\tau_{\max} \geq 0$ be the maximum time lag ts-DAG $\mathcal{G}_{ts}$ and $\mathcal{W} := [t - \tau_{\max}, t]$ be a time window for an arbitrary $t \in \mathbb{Z}$ over the set of discrete time indices. Then, the finite index set $I_{\mathcal{W}} = [m] \times \mathcal{W}$, results in the finite set of vertices $\mathbf{S}_{\mathcal{W}} = \{X_t^i \ : \ (i, t) \in I_{\mathcal{W}}\}$ in the ts-DAG $\mathcal{G}_{ts}$. The induced graph over the set $\mathbf{S}_{\mathcal{W}}$ is defined as the window causal graph (or simply window graph) and denoted by $\mathcal{G}_{\mathcal{W}}$.*

3.2. Causal Discovery with Single versus Multiple Multivariate Time Series

Time series causal discovery is commonly studied in two formats: the *multiple time series* setting, where multiple independent multivariate time series with same underling ts-SCM are observed, and the *sliding-window* setting, where a single dependent multivariate series is converted into data over time windows with a known maximum lag (Runge, 2020; Manten et al., 2025; Wieck-Sosa et al., 2025). In both cases, the central problem is identifying the parents of the contemporaneous slice in the resulting window graph, since Assumption 4 ensures the reconstruction of the time series graph once the window graph is known. While consistency of the Bayesian scoring criterion (BSC), employed as the score in score-based causal discovery, in the multiple time series setting follows relatively directly from standard i.i.d. arguments, it is more delicate in the sliding-window setting due to sample dependence, though recent works have established consistency for the Bayesian information criterion (BIC) under certain assumptions (Bardet et al., 2021). Here, we focus on this shared

window-graph discovery problem while assuming score consistency (Assumption 5, formally stated in Section 5), and present an experimental comparison of the single versus multiple time series data with BIC score in our experimental analysis (Section 6). Our simulations show that the proposed algorithm (Section 4) performs very similarly in both settings.

4. TS-BOSS

4.1. Overview of BOSS

Score-based causal discovery infers a (equivalence class over a) DAG by optimizing a score that trades off goodness-of-fit and complexity; under score consistency and decomposability, improvements in the global score translate into *locally consistent* decisions about adding or removing individual parent-child relationships (see Definition 5 and 6 in (Chickering, 2003) for details on score consistency). Permutation-based methods exploit this structure by searching over variable permutations rather than the full DAG space: for each permutation, one can associate a unique subgraph-minimal DAG consistent with that order (Corollary 2 in (Verma and Pearl, 1990)), reducing the search to identifying an optimal permutation and its induced minimal graph. BOSS operationalizes this in two stages: in Phase 1 it performs a permutation search by repeatedly finding a variable's best position in the ordering and constructing the corresponding DAG efficiently via grow–shrink trees (illustrated in Appendix H, Figures 7 and 8); and in Phase 2 it applies a backward equivalence search (BES, also second phase of GES) to perform edge deletions that improve the score which guarantees asymptotic correctness.

4.2. TS-BOSS Pseudocode and Adaptation Challenges

In the following we present the adaptation idea behind TS-BOSS, and present a pseudocode in Algorithm 1. We note here that the challenges of adaptation to time series and our proposed solution are common to most discrete-search score-based methods and thus their application is not limited to extensions of BOSS. Let $(\mathbf{X}_t)_{t\in\mathbb{Z}}$ denote a stationary multivariate time series with maximum time lag $\tau_{\max}$. To recover the window causal graph up to its Markov equivalence class together with background knowledge of time order, namely the ts-MPDAG of the window causal graph, TS-BOSS proceeds in two phases.

Figure 1 provides a high-level overview of the TS-BOSS workflow; the formal pseudocode is given in Algorithm 1.

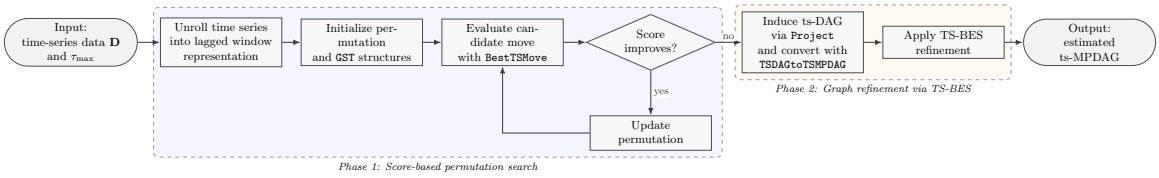


Figure 1: High-level workflow of TS-BOSS. *Phase 1* performs score-based search over admissible contemporaneous permutations. *Phase 2* applies TS-BES to refine the induced graph and obtain the final ts-MPDAG.

First Phase of TS-BOSS. The first phase extends the permutation search and grow–shrink procedure of BOSS via the following modifications:

- **Time-window unrolling.** For a fixed $\tau_{\max}$, the time series is unrolled into the variable set
$$\{\mathbf{X}_{t-\tau_{\max}}, \ldots, \mathbf{X}_t\}, \quad \text{where } \mathbf{X}_t = (X_t^1, \ldots, X_t^m).$$
This yields a window graph with distinct nodes across time steps.

- **Temporal order constraint in permutations** Temporal order is enforced in time series: lagged variables must precede contemporaneous variables. Using the notation of Andrews et al. (2023), the lagged variables must be always placed in early positions than the contemporaneous, permutations take the form
$$\pi = (\underbrace{\mathbf{X}_{t-\tau_{\max}}, \ldots, \mathbf{X}_{t-1}}_{\text{fixed across permutations}}, \mathbf{X}_t).$$

- **Restricted permutation search space.** Under stationarity (Assumption 4), the causal structure is invariant over time. Hence, only edges from lagged variables to $\mathbf{X}_t$ and edges within $\mathbf{X}_t$ must be determined. The permutation search is therefore restricted to contemporaneous variables: only contemporaneous variables are permuted and grow–shrink trees are constructed only for nodes in $\mathbf{X}_t$, while including lagged variables as candidate parents.

We now formalize the first phase of TS-BOSS. Algorithm 1 presents the complete pseudocode, following the structure of BOSS (Andrews et al., 2023) with the time-series adaptations described above. The algorithm relies on several auxiliary subroutines, which we briefly describe for completeness. The subroutine `Unroll` restructures the time series into a lag-unrolled representation, in which the variables $\{X_{t-\tau}^i\}_{\tau=0}^{\tau_{\max}}$ are treated as distinct nodes for scoring and permutation search. The grow–shrink procedure `GST` and the projection step `Project` follow the corresponding routines in BOSS (Andrews et al., 2023): for each contemporaneous node, `GST` incrementally builds and prunes a candidate parent set using local score improvements [1], and `Project` maps the resulting permutation and parent selections to the induced ts-DAG on the lag-unrolled window graph. The routine `BestTSMove` extends the BOSS `best_move` subroutine by restricting the permutation search to admissible contemporaneous positions (see Algorithm 2). Finally, the subroutine `TSDAGtoTSMPDAG` constructs the ts-MPDAG from the induced ts-DAG by applying the procedure described in Appendix B. Since there are finitely many admissible permutations and each accepted move strictly improves the score, Algorithm 1 always terminates.

Second Phase of TS-BOSS In order to prove that the resulting window graph is the most minimal graph that describes the data (i.e. it fulfills Assumption 2 with respect to the time series), the first phase of TS-BOSS must be supplemented with a time series adaptation of the backward equivalence search (TS-BES). TS-BES operates analogously to BES in the static setting with the difference that the neighboring window graphs in the search space correspond to those whose parent sets for the nodes in the contemporaneous slice are fewer.

1. The temporal ordering of lagged variables is fixed, whereas the ordering of contemporaneous variables is arbitrary (see line 2 in Algorithm 1).

Algorithm 1: TS-BOSS (First Phase)

Input: Time series data $\mathbf{D}$, maximum lag $\tau_{\max} \geq 0$
Output: Estimated ts-MPDAG $\widehat{\mathcal{G}_{ts}}$

$\mathbf{D} \leftarrow \texttt{Unroll}(\mathbf{D}, \tau_{\max})$ `// Unroll time series data`
$\pi \leftarrow (\mathbf{X}_{t-\tau_{\max}}, \ldots, \mathbf{X}_{t-1}, \mathbf{X}_t)$ `// Initialize permutation`
$\mathcal{T} \leftarrow \{\texttt{GST}(X_t^i, \widehat{\mathcal{D}})\}_{i=1}^m$ `// Initialize GS trees for` $\mathbf{X}_t$
improved $\leftarrow$ True `// Flag for permutation search`
while improved **do**
  $s_{\text{best}} \leftarrow \mathcal{T}.\texttt{Score}(\pi)$ `// Evaluate current score`
  improved $\leftarrow$ False
  for $i \leftarrow 1$ **to** m **do**
    $\pi' \leftarrow \texttt{BestTSMove}(\mathcal{T}, \pi, X_t^i)$ `// Only contemporaneous variables`
    if $\mathcal{T}.\texttt{Score}(\pi') > s_{\text{best}}$ **then**
      $\pi \leftarrow \pi'$ `// Update permutation`
      improved $\leftarrow$ True
      break
    end
  end
end
$\mathcal{G}_{ts} \leftarrow \mathcal{T}.\texttt{Project}(\pi)$ `// ts-DAG induced by` π
$\widehat{\mathcal{G}_{ts}} \leftarrow \texttt{TSDAGtoTSMPDAG}(\mathcal{G}_{ts})$ `// Compute ts-MPDAG`
return $\widehat{\mathcal{G}_{ts}}$

Algorithm 2: Subroutine: BestTSMove

Input: Grow–shrink trees $\mathcal{T}$, permutation π, variable X_t^i
Output: Updated permutation π

$pos \leftarrow m \cdot \tau_{\max}$ `// Starting index of contemporaneous block`
$s_{\text{best}} \leftarrow \mathcal{T}.\texttt{Score}(\pi)$
$j \leftarrow \pi.\texttt{index}(X_t^i)$ `// Current position of` X_t^i
for $k \leftarrow 1$ **to** m **do**
  $\pi_{temp} \leftarrow \pi.\texttt{move}(X_t^i, pos + k)$ `// Move within contemporaneous block`
  if $\mathcal{T}.\texttt{Score}(\pi_{temp}) > s_{\text{best}}$ **then**
    $s_{\text{best}} \leftarrow \mathcal{T}.\texttt{Score}(\pi_{temp})$
    $j \leftarrow pos + k$
  end
end
$\pi \leftarrow \pi.\texttt{move}(X_t^i, j)$ `// Place at best position`
return π

5. Theoretical Results

We now present results that guarantee correctness of TS-BOSS in the large sample limit.

Assumption 5 *We assume access to:*

(i) a locally consistent Bayesian scoring criterion $\mathcal{B}$ for DAGs and,

(ii) i.i.d. data $\mathbf{D}$ for nodes $\mathbf{S}_\mathcal{W}$ in the window graph $\mathcal{G}_\mathcal{W}$, see Definition 1.

Haughton (1988) has shown that for distributions that are in the curved exponential family, eg. multivariate Gaussian distributions, the Bayesian scoring criterion equals the Bayesian information criterion (BIC) plus a constant error term that becomes insignificant in the large sample limit. We use the BIC score in our TS-BOSS implementation.

In order to employ permutation-search over window causal graph, we will now introduce a few definitions for the time series setting. In the following, we refer to a permutation π over $\mathbf{S}_\mathcal{W}$ as an *admissible permutation* if it respects time order, i.e., for the i^{th} vertex in π called $\pi_i \in \mathbf{S}_\mathcal{W}$, all vertices that precede π_i in π, denoted by $\text{Pre}(i, \pi) = \{\pi_j \ : 1 \leq j < i\}$, have time indices less than or equal to the time index of π_i. Furthermore, the joint probability distribution over the variables $\mathbf{S}_\mathcal{W}$ is denoted by $P_\mathcal{W}$, and it satisfies the *graphoid axioms* (Verma and Pearl, 1990).

Definition 2 (Permutation-induced window graph) *Let $P_\mathcal{W}$ be a graphoid over variables $\mathbf{S}_\mathcal{W}$ associated with the window graph $\mathcal{G}_\mathcal{W}$. Each admissible permutation π of $\mathbf{S}_\mathcal{W}$ induces a DAG $\mathcal{G}^\pi_\mathcal{W}$ over $\mathbf{S}_\mathcal{W}$ defined as follows:*

(i) Let t be the maximal time index of all the variables in π. For each $X^i_t \in \mathbf{S}_\mathcal{W}$ and each $X^j_s \in Pre(X^i_t, \pi)$, if $X^i_t \not\perp\!\!\!\perp X^j_s \mid Pre(X^i_t, \pi) \setminus \{X^j_s\}$, then $X^j_s \to X^i_t$ in $\mathcal{G}^\pi_\mathcal{W}$,

(ii) For all variable indices i, j, time indices $t' < t$ in π and $s' \geq 0$, there exists an edge $X^j_{t'-s'} \to X^i_{t'}$ in $\mathcal{G}^\pi_\mathcal{W}$ if and only if $X^j_{t-s'} \to X^i_t$ was found in (i) above.

Condition (i) in Definition 2 constructs the edges incident into the variables at the maximal time index (also known as the 'contemporaneous slice' in the window graph), and is analogous to the method discussed in (Raskutti and Uhler, 2018). Condition (ii) is needed in order to satisfy time-shift invariance (Assumption 4) that is implicit in the definition of a valid window graph. We use the term *stationary graph* for any window graph that satisfies condition (ii) in Definition 2.

Definition 3 (Window Markov property) *A graphoid $P_\mathcal{W}$ and a window graph $\mathcal{G}_\mathcal{W}$ over variables $\mathbf{S}_\mathcal{W}$ is said to satisfy the window Markov property if for all $X^i_t \in \mathbf{S}_\mathcal{W}$, $X^i_t \perp\!\!\!\perp Nd(X^i_t)_{\mathcal{G}_\mathcal{W}} | Pa(X^i_t)_{\mathcal{G}_\mathcal{W}}$ for all i and the maximal time index t in $\mathcal{G}_\mathcal{W}$. Here $Nd(\cdot)_{\mathcal{G}_\mathcal{W}}$ refers to the set of non-descendants in the graph $\mathcal{G}_\mathcal{W}$.*

From Assumption 1, Assumption 3, Assumption 4 and Definition 3 it follows that if a ts-DAG $\mathcal{G}_{ts}$ satisfies the local Markov property with respect to a multivariate time series, then its corresponding window graph $\mathcal{G}_\mathcal{W}$ satisfies the window Markov property with respect to the graphoid induced over $\mathbf{S}_\mathcal{W}$ and vice-versa.

Definition 4 (Window subgraph minimality) *Let a graph $\mathcal{G}_{\mathcal{W}}$ over vertices $\mathbf{S}_{\mathcal{W}}$ satisfy the window Markov property w.r.t. a graphoid $P_{\mathcal{W}}$. Then $\mathcal{G}_{\mathcal{W}}$ is said to be window subgraph minimal if no stationary subgraph of $\mathcal{G}_{\mathcal{W}}$ satisfies the window Markov property.*

For an overview on subgraph minimality (also known as the minimal I-MAP characterization) in the non-time series setting, see (Pearl, 1988; Verma and Pearl, 1990; Raskutti and Uhler, 2018). Verma and Pearl (1990) proved that, given a permutation, the DAG in which each node's parent set is the smallest subset of its predecessors rendering it conditionally independent of the remaining predecessors satisfies the local Markov property, and that no proper subgraph thereof does (see their Theorem 2 and Corollary 2), also called *subgraph minimality*. Subgraph minimality plays a key role in sparsest-permutation-search algorithms such as that of Raskutti and Uhler (2018), which searches over permutations, assigns to each permutation a subgraph-minimal DAG, and outputs the sparsest such DAG across permutations; its soundness is guaranteed under a strictly weaker counterpart to faithfulness. BOSS uses a similar idea, assigning to each permutation its subgraph-minimal DAG in phase 1 and improving the permutation search through local changes (best-move) and result caching; phase 2 is required for theoretical soundness, since phase 1 does not guarantee that all permutations are searched. Because the subgraph minimality result underlies any sparsest-permutation-based algorithm (including BOSS), extending BOSS to time series requires an analogous result in the time-series setting. Assumption 4 implies that if a window graph satisfies window subgraph minimality, then the corresponding ts-DAG satisfies subgraph minimality with respect to the time-series local Markov property. Thus, we now extend the subgraph minimality to the stationary time-series setting.

Theorem 5 (Permutation-induced window graph minimality) *Let $P_{\mathcal{W}}$ be a graphoid over $\mathbf{S}_{\mathcal{W}}$ and $\mathcal{G}^{\pi}_{\mathcal{W}}$ be a window graph induced from an admissible permutation π. Then $\mathcal{G}^{\pi}_{\mathcal{W}}$ satisfies the window Markov property and is window subgraph minimal.*

All proofs are relegated to Appendix A. In the following result, subgraph minimality for a ts-DAG refers to minimality w.r.t. the time series local Markov property, see explanation after Definition 4.

Lemma 6 *Let assumptions 1-5 be satisfied for data $\mathbf{D}$ and ts-DAG $\mathcal{G}_{ts}$ entailed by a ts-SCM over time series $(\mathbf{X}_t)_{t\in\mathbb{Z}}$, and a Bayesian scoring criterion $\mathcal{B}(\mathcal{G}, \mathbf{D})$ over DAGs $\mathcal{G}$. Then the output graph of phase 1 of TS-BOSS over data $\mathbf{D}$ with score $\mathcal{B}(\mathcal{G}, \mathbf{D})$ satisfies time series local Markov property w.r.t. $(\mathbf{X}_t)_{t\in\mathbb{Z}}$ and is subgraph minimal in the large sample limit.*

This result ensures that phase 1 of TS-BOSS yields the sparsest window graph that is Markovian to the distribution, thus avoiding dense local optima, which is a problem common to score-based algorithms such as GES. Finally, the following result guarantees the asymptotic correctness of TS-BOSS.

Lemma 7 *Let assumptions 1-4 be satisfied for data $\mathbf{D}$ and ts-DAG $\mathcal{G}_{ts}$ resulting from a ts-SCM over time series $(\mathbf{X}_t)_{t\in\mathbb{Z}}$. Let $\widehat{\mathcal{G}}_{ts}$ be a ts-DAG that satisfies the time series local Markov property w.r.t. $(\mathbf{X}_t)_{t\in\mathbb{Z}}$. Then TS-BES with input graph $\widehat{\mathcal{G}}_{ts}$, and a Bayesian scoring criterion $\mathcal{B}(\mathcal{G}, \mathbf{D})$ and time series data $\mathbf{D}$ that satisfies Assumption 5, returns the ts-MPDAG corresponding to the true ts-DAG $\mathcal{G}_{ts}$ in the large sample limit.*

6. Simulation Study

6.1. Data Generation

Synthetic data are generated from multivariate linear ts-SCMs of the form

$$X_t^j := \sum_{X_{t-\tau}^i \in \mathrm{Pa}(X_t^j)} a_{ij}^{(\tau)} X_{t-\tau}^i + \eta_t^j,$$

where $i, j \in [m]$, $\tau \in \{0, \ldots, \tau_{\max}\}$, $a_{ij}^{(\tau)} \in \mathbb{R}$, and η_t^j are mutually independent Gaussian noise terms with zero mean. The contemporaneous graph ($\tau = 0$) is acyclic, while positive-lag self-links allow temporal dependence. Non-stationary parameterizations are discarded, and a burn-in period is removed before collecting samples from the stationary regime.

6.2. Methods

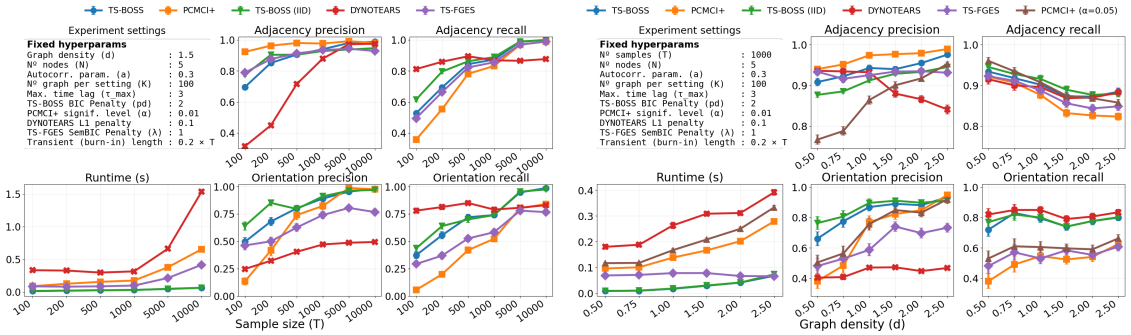
We compare the following methods: (i) **TS-BOSS**: the proposed permutation-based score method adapted to time series by enforcing temporal ordering constraints in the sliding-window setting (see Section 3.2); (ii) **TS-BOSS (i.i.d.)**: TS-BOSS applied in the multiple time series setting (see Section 3.2). (iii) **PCMCI+** (Runge, 2020): a constraint-based method for time-series causal discovery (here applied with the partial correlation test for conditional independence). (iv) **DYNOTEARS** (Pamfil et al., 2020): a score-based continuous optimization method based on linear SCMs with differentiable acyclicity constraints on the contemporaneous structure. (v) **TS-FGES**: a temporal extension of FGES (Ramsey et al., 2017; Chickering, 2003), a more optimized variant of GES (Chickering, 2003), implemented in TETRAD using background knowledge constraints to enforce temporal ordering among variables. For TS-BOSS methods, we present simulations for phase 1, as in (Andrews et al., 2023). PCMCI+, DYNOTEARS and TS-FGES are included for comparison.

6.3. Experimental Setup and Evaluation Metrics

We report adjacency precision and recall, orientation precision and recall, and runtime. Since the estimated graphs are ts-MPDAGs, the ground-truth ts-DAG is first converted into the corresponding ts-MPDAG before evaluation (Appendix B). Details on the extension of the adjacency and orientation metrics of Andrews et al. (2023) to the ts-MPDAG setting are provided in Appendix C. Default values for the data generation process and model parameters are reported in Appendix D. For the experiments in this study, the maximum time lag ($lag_{\max}$) is set to the true lag ($\tau_{\max}$). In real applications, where $\tau_{\max}$ is unknown, $lag_{\max}$ can be tuned using domain knowledge and lag-screening checks (e.g., using the Tigramite package). Unless otherwise specified, all experiments use this configuration. To assess performance, we vary the number of samples, number of nodes, graph density and autocorrelation coefficient across experimental settings. The implementations of the methods and the code used in this work are available at: TS-BOSS repository.

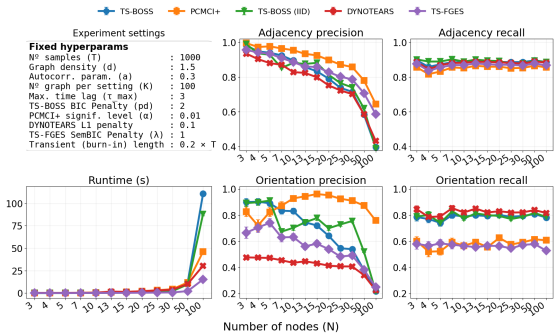
6.4. Results

In Figure 2(a), increasing the sample size T improves precision and recall for most methods. DYNOTEARS achieves high recall already at small T, while PCMCI+ maintains strong

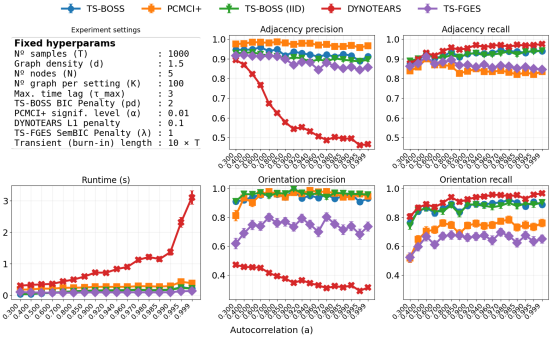


(*a*) Effect of varying sample size (T).

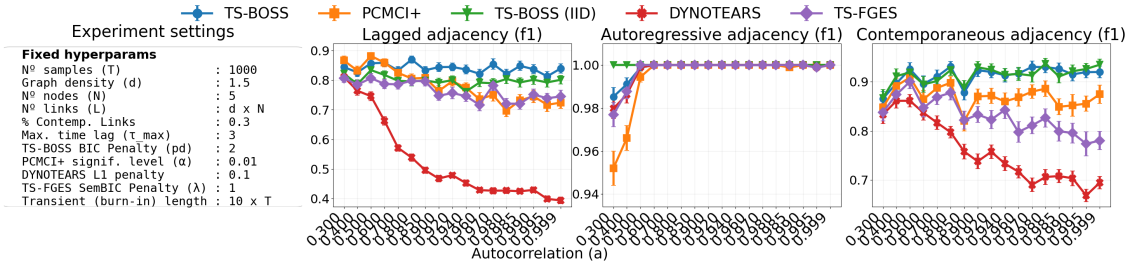
(*b*) Effect of varying graph density (d).



(*c*) Effect of varying number of nodes (N).



(*d*) Effect of varying autocorrelation (a).



(*e*) Effect of varying autocorrelation (a): adjacency F1 by edge type.

Figure 2: Results for the methods in Section 6.2 under the settings defined in Appendix D.

adjacency precision across all settings. TS-BOSS and TS-BOSS (i.i.d.) continue improving for larger T, reaching competitive or superior adjacency and orientation performance while remaining substantially faster than the competing methods. Figure 2(*b*) shows the effect of increasing the graph density d. As d increases, adjacency recall decreases for all methods. Running PCMCI+ with $\alpha = 0.05$ trades precision for recall but still does not outperform TS-BOSS in adjacency recall. DYNOTEARS achieves the highest orientation recall but also the lowest orientation precision. In contrast, TS-BOSS and TS-BOSS (i.i.d.) provide the strongest overall balance between precision and recall while remaining among the fastest methods across most settings. Figure 2(*c*) shows the effect of increasing the number of nodes N. Runtime increases for all methods, particularly for TS-BOSS due to the larger permutation search space. PCMCI+ maintains the strongest precision across

all settings, while DYNOTEARS and both TS-BOSS variants achieve the highest orientation recall, although DYNOTEARS exhibits the lowest orientation precision. TS-BOSS shows a stronger decline in orientation precision for large N. Finally, Figure 2(d) shows the effect of increasing autocorrelation a. PCMCI+ achieves high adjacency precision but decreasing recall, similarly to TS-FGES. DYNOTEARS attains high recall but the weakest precision. In contrast, both TS-BOSS variants remain stable across all values of a and achieve strong adjacency and orientation results. The close agreement between TS-BOSS and TS-BOSS (i.i.d.) across settings also suggests that the BIC score, while not corrected for sliding-window dependence, does not substantially affect performance in the tested regimes.

To complement the precision and recall results, Figure 2(e) reports adjacency F1-scores by edge type as the autocorrelation a varies. Autoregressive edges are recovered almost perfectly by all methods, while for lagged and contemporaneous edges both TS-BOSS variants remain stable as a grows, unlike PCMCI+, TS-FGES and especially DYNOTEARS. Further F1 results, evaluations against the ground-truth ts-DAG and under high autocorrelation are given in Appendix E, and the real-world and memory experiments in Appendix F and Appendix G.

7. Conclusion and Outlook

We present TS-BOSS, an extension of score-based structure learning method BOSS to multivariate time series. We show that TS-BOSS outperforms standard score- and constraint-based algorithms for time series causal discovery. This suggests that permutation-based score search with efficient caching can remain effective when strong sample dependence degrades the behavior of conditional independence testing. We presented theoretical guarantees of TS-BOSS under the i.i.d. window data setting, while illustrating with simulations that non-i.i.d. data from sliding windows over a single multivariate time series does not hamper TS-BOSS performance. In doing so, we established subgraph minimality of the window causal graph obtained using a modified approach to the one presented in Verma and Pearl (1990), thus paving the way for extending sparse permutation-search based methods for causal discovery to the stationary time series setting. However, this stationarity assumption may be restrictive in practice and extending TS-BOSS to non-stationary dynamics remains future work.

An important advantage of constraint-based causal discovery is that it can more directly accommodate violations of orientation faithfulness (Ramsey et al., 2006). Score-based methods, by contrast, may be less transparent in how such violations affect the selected graph, and detecting such violations within the score-based framework would be an avenue for future research. More broadly, the constraint-based and score-based paradigms impose different methodological bottlenecks in single multivariate time series settings. Constraint-based approaches rely on well-calibrated conditional independence tests under temporal dependence of samples, while score-based approaches require theoretical guarantees —e.g., score consistency— tailored to dynamic Bayesian network structure learning. Each set of requirements is nontrivial in practice and motivates further theoretical work. Our ablation studies provide evidence that increasing sample dependence differentially influences the performance of competing approaches and underline the need for further theoretical guarantees in the single multivariate time series setting.

Acknowledgments

We thank Jakob Runge for valuable comments, Paul Prasse for support with using the computational resources provided by the University of Potsdam and anonymous reviewers for helpful comments that improved the clarity of our work.

References

Bryan Andrews, Joseph Ramsey, Rubén Sánchez-Romero, Jazmin Camchong, and Erich Kummerfeld. Fast scalable and accurate discovery of dags using the best order score search and grow-shrink trees. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Charles K. Assaad, Emilie Devijver, and Eric Gaussier. Survey and evaluation of causal discovery methods for time series. *Journal of Artificial Intelligence Research*, 73:767–819, Feb 2022. doi: 10.1613/jair.1.13428. URL https://doi.org/10.1613/jair.1.13428.

Jean-Marc Bardet, Kamila Kare, and William Kengne. Efficient and consistent data-driven model selection for time series. *Submitted to the Annals of Statistics*, 2021. arXiv:2110.09785v1.

Kevin Bello, Bryon Aragam, and Pradeep K. Ravikumar. Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, pages 8226–8239. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/076d5c6c6c9c3a18b5f3e89e3297d1e4-Abstract-Conference.html.

Gustau Camps-Valls, Andreas Gerhardus, Urmi Ninad, Gherardo Varando, Georg Martius, Emili Balaguer-Ballester, Ricardo Vinuesa, Emiliano Diaz, Laure Zanna, and Jakob Runge. Discovering causal relations and equations from data. *Physics Reports*, 1044:1–68, 2023. ISSN 0370-1573. doi: https://doi.org/10.1016/j.physrep.2023.10.005. URL https://www.sciencedirect.com/science/article/pii/S0370157323003411. Discovering causal relations and equations from data.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2003. doi: 10.1162/153244303321897717. URL https://jmlr.org/papers/v3/chickering02b.html.

Juan L. Gamella, Jonas Peters, and Peter Bühlmann. Causal chambers as a real-world physical testbed for AI methodology. *Nature Machine Intelligence*, 2025. doi: 10.1038/s42256-024-00964-x.

Andreas Gerhardus. Characterization of causal ancestral graphs for time series with latent confounders. *The Annals of Statistics*, 52(1):103–130, February 2024. doi: 10.1214/23-AOS2262.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10:524, 2019. doi: 10.3389/fgene.2019.00524.

Dominique M. A. Haughton. On the choice of a model to fit data from an exponential family. *Annals of Statistics*, 16(1):342–355, 1988. doi: 10.1214/aos/1176350709. URL https://doi.org/10.1214/aos/1176350709.

Tom Hochsprung, Jakob Runge, and Andreas Gerhardus. A global markov property for solutions of stochastic difference equations and the corresponding full time graphs. In *The 40th Conference on Uncertainty in Artificial Intelligence*, 2024. URL https://openreview.net/forum?id=o5pOj2IDYB.

Aapo Hyvärinen, Kun Zhang, Shohei Shimizu, and Patrik O. Hoyer. Estimation of a structural vector autoregression model using non-gaussianity. *Journal of Machine Learning Research*, 11:1709–1731, 2010. URL https://www.jmlr.org/papers/v11/hyvarinen10a.html.

Wai-Yin Lam, Bryan Andrews, and Joseph Ramsey. Greedy relaxations of the sparsest permutation algorithm. In James Cussens and Kun Zhang, editors, *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pages 1052–1062. PMLR, 01–05 Aug 2022. URL https://proceedings.mlr.press/v180/lam22a.html.

Daniel Malinsky and Peter Spirtes. Causal structure learning from multivariate time series in settings with unmeasured confounding. In Thuc Duy Le, Kun Zhang, Emre Kıcıman, Aapo Hyvärinen, and Lin Liu, editors, *Proceedings of 2018 ACM SIGKDD Workshop on Causal Disocvery*, volume 92 of *Proceedings of Machine Learning Research*, pages 23–47. PMLR, 20 Aug 2018. URL https://proceedings.mlr.press/v92/malinsky18a.html.

Georg Manten, Cecilia Casolo, Søren Wengel Mogensen, and Niki Kilbertus. An asymmetric independence model for causal discovery on path spaces. In Biwei Huang and Mathias Drton, editors, *Proceedings of the Fourth Conference on Causal Learning and Reasoning*, volume 275 of *Proceedings of Machine Learning Research*, pages 64–89. PMLR, 07–09 May 2025. URL https://proceedings.mlr.press/v275/manten25a.html.

Christopher Meek. Causal inference and causal explanation with background knowledge, 2013. URL https://arxiv.org/abs/1302.4972.

Ignavier Ng, Biwei Huang, and Kun Zhang. Structure learning with continuous optimization: A sober look and beyond. In Francesco Locatello and Vanessa Didelez, editors, *Proceedings of the Third Conference on Causal Learning and Reasoning*, volume 236 of *Proceedings of Machine Learning Research*, pages 71–105. PMLR, 01–03 Apr 2024. URL https://proceedings.mlr.press/v236/ng24a.html.

Roxana Pamfil, Nisara Sriwattanaworachai, Shaan Desai, Philip Pilgerstorfer, Konstantinos Georgatzis, Paul Beaumont, and Bryon Aragam. Dynotears: Structure learning from time-series data. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume

108 of *Proceedings of Machine Learning Research*, pages 1595–1605. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/pamfil20a.html.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Series in Representation and Reasoning. Morgan Kaufmann, San Mateo, CA, 1988. ISBN 1558604790.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition, 2009. ISBN 052189560X. doi: 10.1017/CBO9780511803161.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. The MIT Press, 2017. ISBN 9780262037310.

Joseph Ramsey, Peter Spirtes, and Jiji Zhang. Adjacency-faithfulness and conservative causal inference. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'06, page 401–408, Arlington, Virginia, USA, 2006. AUAI Press. ISBN 0974903922.

Joseph Ramsey, Madelyn Glymour, Ruben Sanchez-Romero, and Clark Glymour. 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(2):121–129, 2017.

Garvesh Raskutti and Caroline Uhler. Learning directed acyclic graph models based on sparsest permutations. *Stat*, 7(1):e183, 2018. doi: https://doi.org/10.1002/sta4.183. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.183. e183 sta4.183.

Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! causal discovery benchmarks may be easy to game. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 27772–27784. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/e987eff4a7c7b7e580d659feb6f60c1a-Paper.pdf.

Jakob Runge. Discovering contemporaneous and lagged causal relations in autocorrelated nonlinear time series datasets. In *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2020.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, Jonas Peters, Rick Quax, Markus Reichstein, Marten Scheffer, Bernhard Schölkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in Earth system sciences. *Nature Communications*, 10(1):2553, June 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-10105-3. URL https://www.nature.com/articles/s41467-019-10105-3. Number: 1 Publisher: Nature Publishing Group.

Liam Solus, Yu Wang, and Caroline Uhler. Consistency guarantees for greedy permutation-based causal inference algorithms. *Biometrika*, 108(4):795–814, 2021. doi: 10.1093/biomet/asaa104.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT Press, 2 edition, 2000.

Chandler Squires and Caroline Uhler. Causal structure learning: A combinatorial perspective. *Foundations of Computational Mathematics*, 23(5):1781–1815, 2023. doi: 10.1007/s10208-022-09581-9. URL https://doi.org/10.1007/s10208-022-09581-9.

Thomas Verma and Judea Pearl. Causal networks: Semantics and expressiveness. In Ross D. Shachter, Tod S. Levitt, Laveen N. Kanal, and John F. Lemmer, editors, *Uncertainty in Artificial Intelligence*, volume 9 of *Machine Intelligence and Pattern Recognition*, pages 69–76. North-Holland, 1990. doi: https://doi.org/10.1016/B978-0-444-88650-7.50011-1. URL https://www.sciencedirect.com/science/article/pii/B9780444886507500111.

Michael Wieck-Sosa, Michel F. C. Haddad, and Aaditya Ramdas. Conditional independence testing with a single realization of a multivariate nonstationary nonlinear time series, 2025. URL https://arxiv.org/abs/2504.21647.

Marcel Wienöbst, Leonard Henckel, and Sebastian Weichwald. Embracing discrete search: A reasonable approach to causal structure learning, 2025. URL https://arxiv.org/abs/2510.04970.

Xun Zheng, Bryon Aragam, Pradeep K. Ravikumar, and Eric P. Xing. Dags with no tears: Continuous optimization for structure learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, pages 9472–9483. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/hash/e347c51419ffb23ca3fd5050202f9c3d-Abstract.html.

Appendix A. Proofs

In this section, we present proofs for theoretical results presented in Section 5. First we recall the following definition from (Lam et al., 2022) that is needed for the proofs.

Definition 8 (RU projection) *Let P denote a graphoid over a set of vertices $\mathbf{V}$, and let $\Pi(\mathbf{V})$ denote the set of all permutations over the vertices $\mathbf{V}$. For the i^{th} vertex in $\pi \in \Pi(\mathbf{V})$ called $\pi_i \in \mathbf{S}_{\mathcal{W}}$, all vertices that precede π_i in π are denoted by $Pre(i, \pi) = \{\pi_j \ : 1 \leq j < i\}$. We use X_i to denote the random variable corresponding to a vertex π_i and $\mathbf{X}_{Pre(i,\pi)}$ to denote the set of random variables corresponding to the vertex set $Pre(i, \pi)$. Then, each $\pi \in \Pi(\mathbf{V})$ induced a DAG $\mathcal{G}_\pi$ satisfying the following condition: An edge $\pi_j \to \pi_k$ exists in $\mathcal{G}_\pi$ if and only if $\pi_j \in Pre(k, \pi)$ and $X_j \not\perp\!\!\!\perp_{\mathcal{P}} X_k | \mathbf{X}_{Pre(k,\pi)\setminus j}$.*

Theorem 5 (Permutation-induced window graph minimality) Let $P_{\mathcal{W}}$ denote a graphoid over $\mathbf{S}_{\mathcal{W}}$ and $\mathcal{G}^{\pi}_{\mathcal{W}}$ be a window graph induced from an admissible permutation π. Then $\mathcal{G}^{\pi}_{\mathcal{W}}$ satisfies the window Markov property and is window subgraph minimal.

Proof For the permutation π one can construct the induced DAG following the strategy *RU* (see Definition 8) defined in (Lam et al., 2022) and motivated from (Raskutti

and Uhler, 2018), without distinguishing between time indices. It follows that the graph $\mathcal{G}_{RU} = RU(\pi)$ is such that $\mathcal{G}^{\pi}_{\mathcal{W}} \subseteq \mathcal{G}_{RU}$, where there may be additional edges in $\mathcal{G}_{RU}$ due to common confounders of window variables $\mathbf{S}_{\mathcal{W}}$ lying outside the window. Additionally, for each i and the maximal time index t in π, $\mathrm{Pa}(X^i_t)_{\mathcal{G}_{RU}} = \mathrm{Pa}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}}$, because the edges incident into the contemporaneous slice have been found using the same strategy in both methods. Further $\mathrm{Nd}(X^i_t)_{\mathcal{G}_{RU}} = \mathrm{Nd}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}}$ because the permutation π was admissible and thus all variables preceding the variables in the contemporaneous slice cannot be their descendants. From Theorem 2 in (Verma and Pearl, 1990), it follows that $\mathcal{G}_{RU}$ satisfies $X^i_t \perp\!\!\!\perp \mathrm{Nd}(X^i_t)_{\mathcal{G}_{RU}} \mid \mathrm{Pa}(X^i_t)_{\mathcal{G}_{RU}} \Rightarrow X^i_t \perp\!\!\!\perp \mathrm{Nd}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}} \mid \mathrm{Pa}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}}$. Thus, $\mathcal{G}^{\pi}_{\mathcal{W}}$ satisfies window Markov property.

For the subgraph minimality proof, assume the contrary, namely that there exists a stationary subgraph $\mathcal{H}^{\pi}_{\mathcal{W}} \subset \mathcal{G}^{\pi}_{\mathcal{W}}$ that satisfies the window Markov property. Stationarity of the graph implies that $\mathrm{Pa}(X^i_t)_{\mathcal{H}^{\pi}_{\mathcal{W}}} \subset \mathrm{Pa}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}}$. Let $\mathbf{D}^i_t := \mathrm{Pa}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}} \setminus \mathrm{Pa}(X^i_t)_{\mathcal{H}^{\pi}_{\mathcal{W}}}$. For all $X^j_{t'} \in \mathbf{D}^i_t$, $X^j_{t'} \in \mathrm{Nd}(X^i_t)_{\mathcal{H}^{\pi}_{\mathcal{W}}}$ because if it were a descendant of X^i_t in $\mathcal{H}^{\pi}_{\mathcal{W}}$, it would also be a descendant of X^i_t in $\mathcal{G}^{\pi}_{\mathcal{W}}$ which would violate acyclicity. From window Markovianity of $\mathcal{H}^{\pi}_{\mathcal{W}}$ we have $\mathbf{D}^i_t \perp\!\!\!\perp X^i_t \mid \mathrm{Pa}(X^i_t)_{\mathcal{H}^{\pi}_{\mathcal{W}}}$. By the weak union property of the graphoid axioms we have $X^j_{t'} \perp\!\!\!\perp X^i_t \mid (\mathrm{Pa}(X^i_t)_{\mathcal{H}^{\pi}_{\mathcal{W}}} \cup (\mathbf{D}^i_t \setminus X^j_{t'}))$ which is equivalent to $X^j_{t'} \perp\!\!\!\perp X^i_t \mid (\mathrm{Pa}(X^i_t)_{\mathcal{G}^{\pi}_{\mathcal{W}}} \setminus X^j_{t'})$. However, this would contradict condition (i) in definition 2. Therefore, $\mathcal{G}^{\pi}_{\mathcal{W}}$ must be window subgraph minimal. ■

Lemma 6 Let assumptions 1-5 be satisfied for data $\mathbf{D}$ and ts-DAG $\mathcal{G}_{ts}$ resulting from a ts-SCM over time series $(\mathbf{X}_t)_{t\in\mathbb{Z}}$, and a Bayesian scoring criterion $\mathcal{B}(\mathcal{G}, \mathbf{D})$ over DAGs. Then the output graph of phase 1 of TS-BOSS over data $\mathbf{D}$ with score $\mathcal{B}(\mathcal{G}, \mathbf{D})$ satisfies time series local Markov property w.r.t. $(\mathbf{X}_t)_{t\in\mathbb{Z}}$ and is subgraph minimal in the large sample limit.

Proof Suppose not, then the output ts-DAG $\widehat{\mathcal{G}_{ts}}$ of phase 1 of TS-BOSS is such that there exists an $X^j_s \in \mathrm{Nd}(X^i_t)_{\widehat{\mathcal{G}_{ts}}}$ such that $X^i_t \not\perp\!\!\!\perp X^j_s \mid \mathrm{Pa}(X^i_t)_{\widehat{\mathcal{G}_{ts}}}$. From Assumption 1, Assumption 3 and Assumption 4, it follows that there exists a variable that is a non-descendant in the window graph and violates the window Markov property. W.l.o.g. let this variable be X^j_s. Consider the last permutation in phase 1 before the permutation search terminates because no score improvement can be found. Local score consistency of $\mathcal{B}(\mathcal{G}, \mathbf{D})$, i.e. Assumption 5, implies that the ts-DAG that results from adding edge $X^j_s \to X^i_t$ when computing the permutation-induced graph, has a higher score than for a ts-DAG without this edge, therefore the first phase cannot terminate at $\widehat{\mathcal{G}_{ts}}$. ■

Lemma 7 Let assumptions 1-4 be satisfied for data $\mathbf{D}$ and ts-DAG $\mathcal{G}_{ts}$ resulting from a ts-SCM over time series $(\mathbf{X}_t)_{t\in\mathbb{Z}}$. Let $\widehat{\mathcal{G}}_{ts}$ be a ts-DAG that satisfies the time series local Markov property w.r.t. $(\mathbf{X}_t)_{t\in\mathbb{Z}}$. Then TS-BES with input graph $\widehat{\mathcal{G}}_{ts}$, and a Bayesian scoring criterion $\mathcal{B}(\mathcal{G}, \mathbf{D})$ and time series data $\mathbf{D}$ that satisfies Assumption 5, returns the ts-MPDAG corresponding to the true ts-DAG $\mathcal{G}_{ts}$ in the large sample limit.

Proof TS-BES takes as input the graph $\widehat{\mathcal{G}}_{ts}$. Let $\widehat{\mathcal{G}_{\mathcal{W}}}$ be the corresponding window graph and $\mathcal{G}_{\mathcal{W}}$ be the true window graph. Suppose $\mathcal{G}_{\mathcal{W}} \subset \widehat{\mathcal{G}_{\mathcal{W}}}$. For both these graphs consider only the edges incident into the contemporaneous slice because TS-BES attempts deletion of these edges only by Assumption 3 and Assumption 4. Following the proof of soundness

of BES in (Chickering, 2003), we note that Theorem 4 therein, namely the Meek conjecture, implies that the the true $\mathcal{G}_{\mathcal{W}}$ can be obtained from the estimated $\widehat{\mathcal{G}_{\mathcal{W}}}$ with a sequence of edge deletions and covered edge reversals. Consider the last graph $\mathcal{H}$ in the sequence from $\mathcal{G}_{\mathcal{W}}$ to $\widehat{\mathcal{G}_{\mathcal{W}}}$ which has fewer edges than $\widehat{\mathcal{G}_{\mathcal{W}}}$. By Assumption 5, $\mathcal{H}$ has a higher score than $\widehat{\mathcal{G}_{\mathcal{W}}}$, thus TS-BES cannot terminate at $\widehat{\mathcal{G}_{\mathcal{W}}}$. ■

Appendix B. Construction of the ts-MPDAG from a ts-DAG

Given a ts-DAG, the corresponding ts-MPDAG is constructed as follows:

1. All edges are first rendered unoriented.
2. The orientation of all lagged edges is then fixed, since temporal order uniquely determines their direction.
3. All colliders are oriented, including both contemporaneous colliders and mixed-time colliders. A mixed-time collider refers to a v-structure of the form
$$X^i_{t-\tau} \to Y^j_t \leftarrow Z^k_t,$$
where one parent is lagged and the other contemporaneous.
4. Meek's rule R1 (Meek, 2013) is applied to unshielded triples involving both lagged and contemporaneous edges, preventing the introduction of additional mixed-time colliders.
5. Finally, Meek's rules R1–R3 (Meek, 2013) are applied to the contemporaneous subgraph until no further orientations are possible. Note that R2 and R3 only need to be applied to the contemporaneous subgraph, because no subgraph satisfying the precondition for R2 and R3 contains mixed-time triples.

Appendix C. Evaluation Metrics

The evaluation protocol introduced in (Andrews et al., 2023) computes performance metrics by comparing the estimated graphs produced by the evaluated methods to the corresponding ground-truth DAGs. In this work, since the estimated graphs are ts-MPDAGs, the ground-truth ts-DAG is first converted into the corresponding ts-MPDAG (Appendix B), and all metrics are computed at the ts-MPDAG level. The evaluation protocol described in this section extends the adjacency and orientation metrics of (Andrews et al., 2023) to account for undirected edges in ts-MPDAGs.

As mentioned in Section 6.3, we report adjacency precision and recall, orientation precision and recall, and runtime. Precision and recall are computed as

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad \text{Recall} = \frac{TP}{TP + FN}.$$

The definitions of true positives (TP), false positives (FP), false negatives (FN) and true negatives (TN) differ for adjacency and orientation evaluation and are described in the following subsections.

C.1. Adjacency Evaluation

Adjacency precision and recall are computed by evaluating whether an edge exists between two nodes independently of its orientation. Since adjacency evaluation depends only on edge existence, the presence of undirected edges in ts-MPDAGs does not modify the DAG-based evaluation protocol introduced in (Andrews et al., 2023). The corresponding adjacency evaluation outcomes are summarized in Table 1.

C.2. Orientation Evaluation

For orientation evaluation, we adopt the arrowhead-based criterion introduced in (Andrews et al., 2023). For each ordered pair $(X^i_{t-\tau}, X^j_{t-\tau})$ with $\tau = 0$, orientation is evaluated locally by checking whether the edge between X^i_t and X^j_t contains an arrowhead into node X^i_t. Orientation evaluation is therefore restricted to contemporaneous edges, since lagged edges are uniquely oriented by temporal order. The arrowhead check is performed separately for each node of every edge.

Let $\mathbb{I}_{\to i}(j)$ denote the indicator function defined as

$$\mathbb{I}_{\to i}(j) = \begin{cases} 1 & \text{if the edge between } X^i_t \text{ and } X^j_t \text{ contains an arrowhead into } X^i_t, \\ 0 & \text{otherwise.} \end{cases}$$

The orientation evaluation outcomes and the corresponding indicator values for the ground-truth ts-MPDAG and the estimated ts-MPDAG are summarized in Table 1, where $\mathbb{I}^{\text{true}}_{\to i}(j)$ denotes the indicator value in the ground-truth graph and $\mathbb{I}^{\text{est}}_{\to i}(j)$ the corresponding value in the estimated graph.

If the ground-truth graph contains an undirected edge $X^i_t \circ\!\!-\!\!\circ X^j_t$, then $\mathbb{I}^{\text{true}}_{\to i}(j) = 0$, since no arrowhead into X^i_t is present. Consequently, predicting no arrowhead into X^i_t is counted as correct.

Appendix D. Default Experimental Parameters

Table 2 reports the default parameters used for the data generation process defined in Section 6.1 and for the models compared in this work (see Section 6.2), unless otherwise specified.

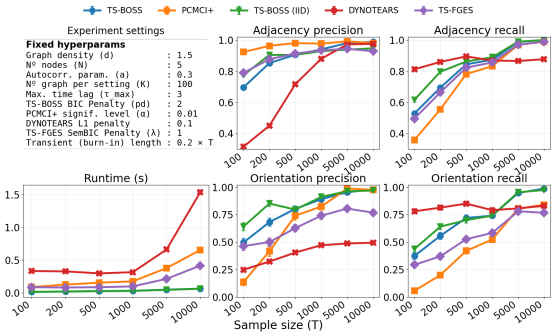
Appendix E. Further Simulations

E.1. Results Compared with the Ground-Truth TS-DAG

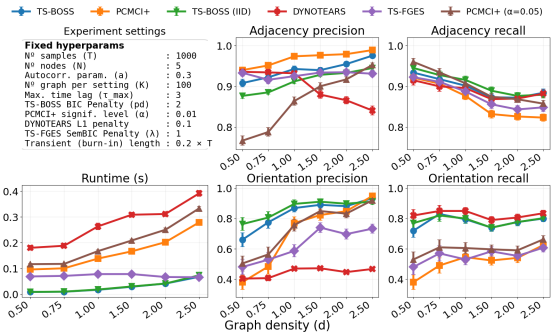
To mirror the evaluation protocol used in (Andrews et al., 2023), we additionally report results for the experiments presented in Section 6, comparing the ground-truth ts-DAG with the ts-MPDAG returned by each method. The results are shown in Figure 3.

True edge	Estimated edge	Adjacency	$\mathbb{I}^{\text{true}}_{\to i}(j)$	$\mathbb{I}^{\text{est}}_{\to i}(j)$	Orientation
$X_t^i \leftarrow X_t^j$	$X_t^i \leftarrow X_t^j$	TP	1	1	TP
$X_t^i \leftarrow X_t^j$	$X_t^i \rightarrow X_t^j$	TP	1	0	FN
$X_t^i \leftarrow X_t^j$	$X_t^i \circ\!\!-\!\!\circ X_t^j$	TP	1	0	FN
$X_t^i \leftarrow X_t^j$	$X_t^i \dots X_t^j$	FN	1	0	FN
$X_t^i \rightarrow X_t^j$	$X_t^i \leftarrow X_t^j$	TP	0	1	FP
$X_t^i \rightarrow X_t^j$	$X_t^i \rightarrow X_t^j$	TP	0	0	TN
$X_t^i \rightarrow X_t^j$	$X_t^i \circ\!\!-\!\!\circ X_t^j$	TP	0	0	TN
$X_t^i \rightarrow X_t^j$	$X_t^i \dots X_t^j$	FN	0	0	TN
$X_t^i \circ\!\!-\!\!\circ X_t^j$	$X_t^i \leftarrow X_t^j$	TP	0	1	FP
$X_t^i \circ\!\!-\!\!\circ X_t^j$	$X_t^i \rightarrow X_t^j$	TP	0	0	TN
$X_t^i \circ\!\!-\!\!\circ X_t^j$	$X_t^i \circ\!\!-\!\!\circ X_t^j$	TP	0	0	TN
$X_t^i \circ\!\!-\!\!\circ X_t^j$	$X_t^i \dots X_t^j$	FN	0	0	TN
$X_t^i \dots X_t^j$	$X_t^i \leftarrow X_t^j$	FP	0	1	FP
$X_t^i \dots X_t^j$	$X_t^i \rightarrow X_t^j$	FP	0	1	FP
$X_t^i \dots X_t^j$	$X_t^i \circ\!\!-\!\!\circ X_t^j$	FP	0	0	–
$X_t^i \dots X_t^j$	$X_t^i \dots X_t^j$	TN	0	0	–

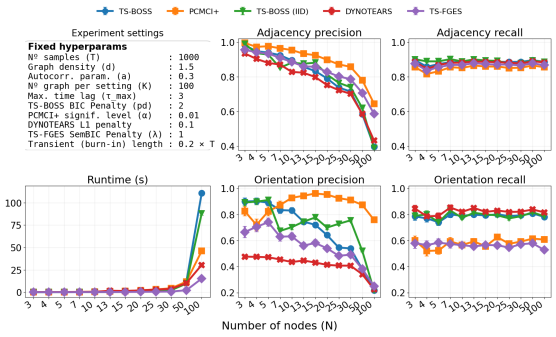
Table 1: Adjacency and orientation evaluation outcomes for ts-MPDAG comparison.



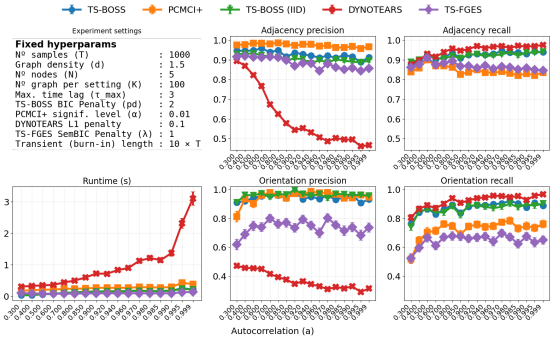
(a) Effect of varying sample size (T).



(b) Effect of varying graph density (d).



(c) Effect of varying number of nodes (N).



(d) Effect of varying autocorrelation (a).

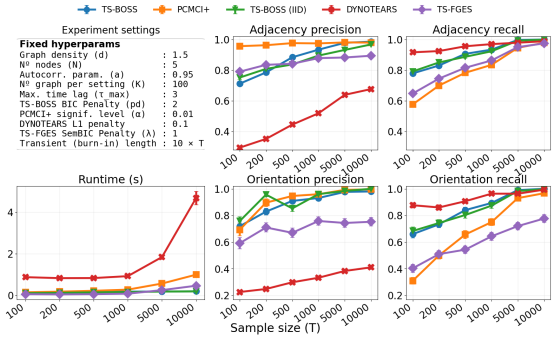
Figure 3: Results comparing the ground-truth ts-DAG to the ts-MPDAG returned by each method under the experimental settings defined in Section 6.

Parameter	Default value
Data generation	
Number of variables (nodes) N	5
Sample size T	1000
Number of graphs per setting K	100
Coefficient distribution	discrete in $[-0.5, 0.5]$
Graph density d	1.5
Number of links L	$\lfloor d \times N \rfloor$
% of contemporaneous links	0.3
Autocorrelation parameter a	0.3
Autocorrelation sampling	$U([\max(0.1, a - 0.3),\ a])$
Coupling function family	linear
Maximum true time lag $\tau_{\max}$	3
Transient (burn-in) length	$0.2 \times T$
Nonstationary realizations	discarded
TS-BOSS	
Maximum time lag $lag_{\max}$	$\tau_{\max}$
Score function	BIC
BIC penalty weight w	$\frac{pd}{4} \cdot \frac{\log(T)}{T}$
BIC penalty discount pd	2
PCMCI+	
Maximum time lag $lag_{\max}$	$\tau_{\max}$
Significance level	$\alpha = 0.01$
Contemporaneous rule	Majority
Conditional test	ParCorr ($significance = analytic$)
Minimum lag $\tau_{\min}$	0
TS-FGES	
Maximum time lag $lag_{\max}$	$\tau_{\max}$
Score function	SemBicScore
Penalty discount	1.0
Faithfulness assumption	True
Symmetric first step	False
Replicating	True
Maximum degree	-1
Number of threads	1
DYNOTEARS	
Maximum time lag $lag_{\max}$	$\tau_{\max}$
lambda_w	0.1
lambda_a	0.1
Weight threshold	10^{-4}
Maximum iterations	100
Tolerance h_{tol}	10^{-8}

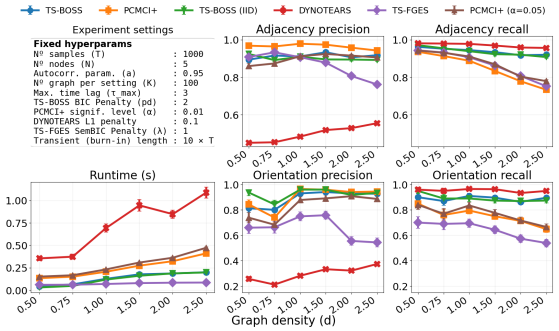
Table 2: Default parameter settings used throughout the simulation study in Section 6.

E.2. Results under High Autocorrelation

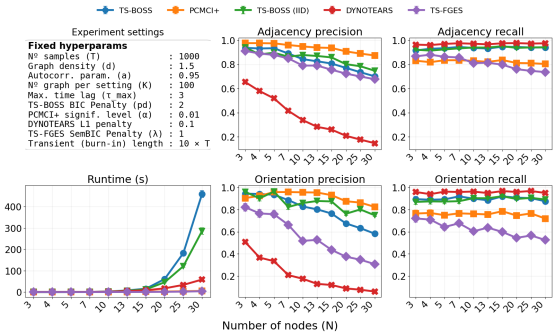
We further repeat the experiments described in Section 6 under a fixed autocorrelation parameter of $a = 0.95$. The results are shown in Figure 4.



(a) Effect of varying sample size (T).



(b) Effect of varying graph density (d).

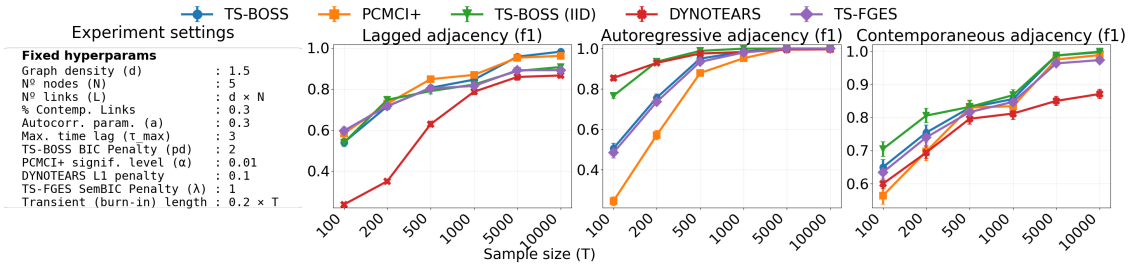


(c) Effect of varying number of nodes (N).

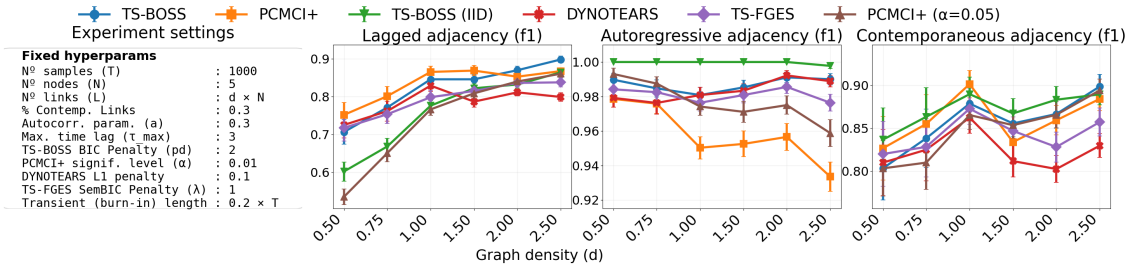
Figure 4: Results under the experimental settings defined in Section 6 with fixed autocorrelation $a = 0.95$.

E.3. Adjacency Results by Edge Type

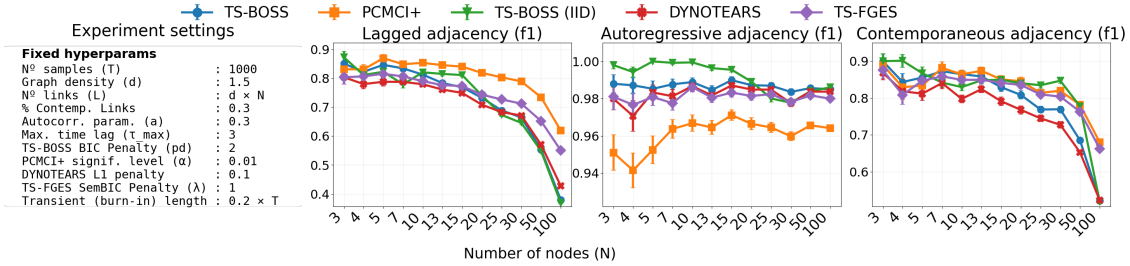
For the experimental settings defined in Section 6, we further report adjacency performance separately for contemporaneous, lagged and autoregressive edges. The results are shown in Figure 5.



(*a*) Effect of varying sample size (T).



(*b*) Effect of varying graph density (d).



(*c*) Effect of varying number of nodes (N).

Figure 5: Adjacency F1-scores reported separately for lagged, autoregressive and contemporaneous edges under the experimental settings defined in Section 6.

Appendix F. Real-World Evaluation on Causal Chamber

The synthetic evaluation is complemented with a real-world experiment using the Causal Chamber wind-tunnel dataset (Gamella et al., 2025). The experiment uses the 16 independent actuator random walks from `wt_walks_v1`, processing and evaluating each run separately.

Each run contains time-series measurements of the same 16 variables, which constitute the nodes of the causal graph. Within each trajectory, every variable is independently

rescaled to $[0, 1]$ by min–max normalization. This places measurements with different physical units on a comparable numerical scale and improves numerical stability. The same preprocessing is applied to all methods.

TS-BOSS is compared with PCMCI+, DYNOTEARS and TS-FGES. All methods use a maximum lag of $\tau_{\max} = 10$; the remaining settings follow the default configurations summarized in Table 2.

The Causal Chamber provides a validated directed causal graph between variables, but it does not assign temporal lags to its edges. Exact lag recovery therefore cannot be evaluated. For evaluation, each estimated temporal relation is projected onto a directed static relation:

$$X^i_{t-\tau} \to X^j_t \quad \mapsto \quad X_i \to X_j.$$

Relations estimated at multiple lags are counted once, and autoregressive links are excluded because they are not represented in the reference graph. The projected graphs are evaluated using the adjacency metrics described in Appendix C.

Table 3 reports the mean and standard error across the 16 trajectories, with SE $= s/\sqrt{16}$. TS-BOSS achieves the highest mean adjacency recall and F1 score, whereas TS-FGES obtains the highest precision and a closely comparable F1 score. Thus, TS-BOSS recovers a larger proportion of the validated physical causal relations, at the cost of a higher number of false-positive adjacencies. Because the reference graph contains no lag annotations, these results measure variable-level causal-structure recovery rather than exact temporal-lag recovery. The complete experiment is available in the TS-BOSS repository.

Method	Precision	Recall	F1 score
TS-BOSS	0.311 ± 0.008	$\mathbf{0.630 \pm 0.016}$	$\mathbf{0.416 \pm 0.010}$
TS-FGES	$\mathbf{0.395 \pm 0.021}$	0.433 ± 0.024	0.410 ± 0.021
PCMCI+	0.318 ± 0.020	0.276 ± 0.016	0.293 ± 0.015
DYNOTEARS	0.253 ± 0.015	0.113 ± 0.014	0.153 ± 0.014

Table 3: Static adjacency recovery on the Causal Chamber wind-tunnel dataset. Values are mean ± standard error across 16 independent runs.

Appendix G. Memory Scalability

We further evaluate memory consumption as the number of nodes increases, while keeping the remaining experimental settings fixed according to the default configuration in Table 2. For each value of N, we generate 10 independent causal graphs. Within each repetition, all time-series methods are evaluated on the same data realization, while TS-BOSS (i.i.d.) uses the corresponding i.i.d. realization generated from the same causal model.

Each method is executed in an isolated process, and memory consumption is measured as its peak resident set size (RSS). This includes dependencies, native allocations, and the JVM used by TS-FGES. The experiments were conducted on an Apple M5 Pro system with 48 GB of RAM, using Python 3.11.16 and OpenJDK 26.0.2.

As shown in Figure 6, TS-BOSS maintains the lowest peak memory consumption among the methods across the evaluated dimensions. At $N = 100$, TS-BOSS requires 111.2 MiB, compared with 170.7 MiB for PCMCI+, 198.5 MiB for DYNOTEARS, and 706.7 MiB for TS-FGES. Overall, the peak memory consumption of TS-BOSS increases moderately over the evaluated range of node counts.

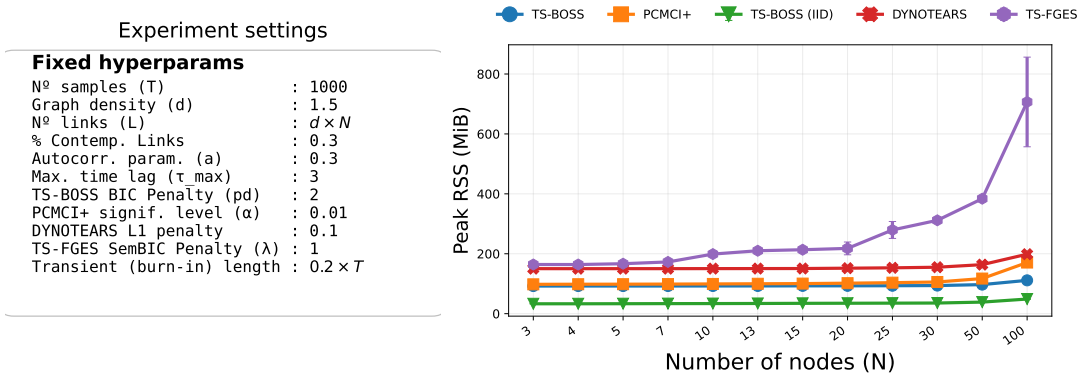


Figure 6: Memory consumption as the number of nodes increases. Points represent the mean peak RSS over 10 independent causal graphs, and error bars denote ± one sample standard deviation.

Appendix H. Grow and Shrink Algorithm Overview

The following figures illustrate the grow and shrink phases of the GST procedure used in BOSS (Andrews et al., 2023) and inherited by TS-BOSS.

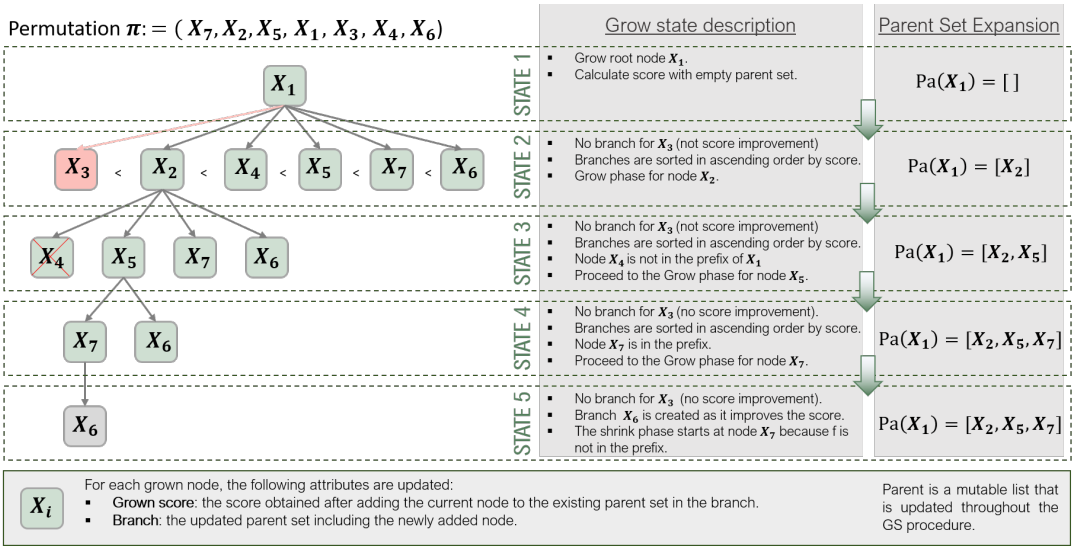


Figure 7: An overview of the `grow` algorithm presented in (Andrews et al., 2023). Given a target variable X_1 and a permutation π, the grow phase builds X_1's candidate parent set by evaluating each available variable as a possible addition to the parent set and constructing branches on those that strictly improve X_1's score. It then sorts these branches by the resulting (post-addition) grow score and chooses the best-ranked branch permissible by the order of π, that variable is added as a parent; the procedure continues until no further improving variable from the prefix of π can be added to X_1's parent set.

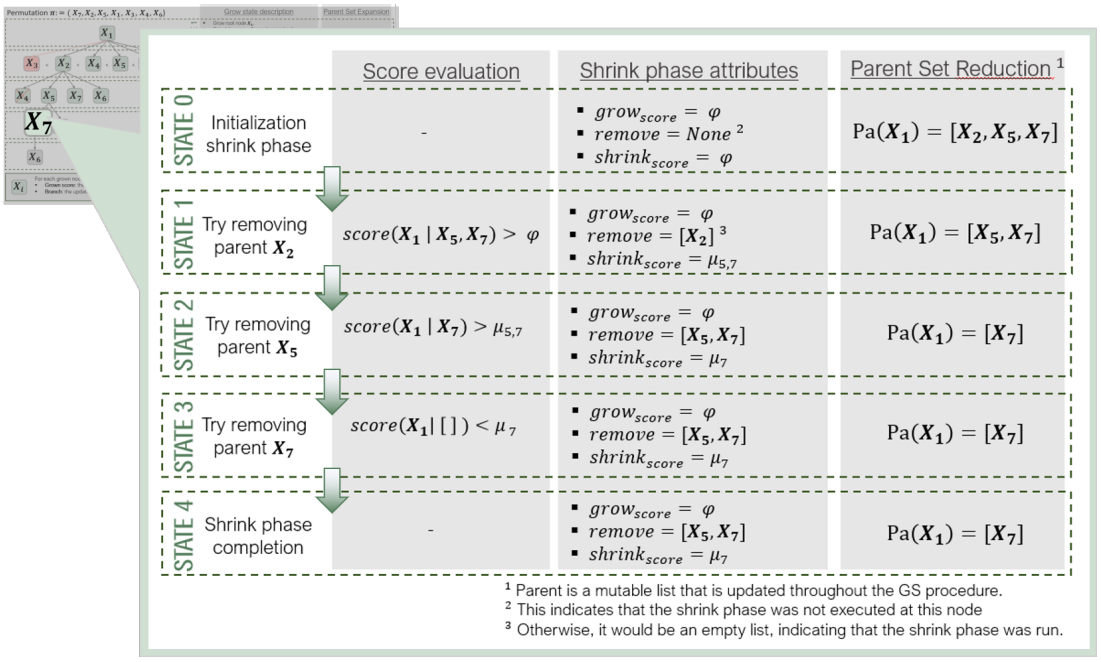


Figure 8: An overview of the `shrink` algorithm presented in (Andrews et al., 2023). After the grow phase, the parent set can include redundant variables, for instance, those that became redundant once other parents were added. The shrink phase prunes this set by iteratively removing any parent whose deletion improves the score, repeating until no further score-improving removals are possible.