Causal screening for dynamical systems

Søren Wengel Mogensen
Department of Mathematical Sciences
University of Copenhagen
Copenhagen, Denmark
swengel@math.ku.dk

Abstract

Many classical algorithms output graphical representations of causal structures by testing conditional independence among a set of random variables. In dynamical systems, local independence can be used analogously as a testable implication of the underlying data-generating process. We suggest some inexpensive methods for causal screening which provide output with a sound causal interpretation under the assumption of ancestral faithfulness. The popular model class of linear Hawkes processes is used to provide an example of a dynamical causal model. We argue that for sparse causal graphs the output will often be close to complete. We give examples of this framework and apply it to a challenging biological system.

1 Introduction

Constraint-based causal learning is computationally and statistically challenging. There is a large literature on learning structures that are represented by directed acyclic graphs (DAGs) or marginalizations thereof (see e.g. [10] for references). The fast causal inference algorithm, FCI, [19] provides in a certain sense maximally informative output [23], but at the cost of using a large number of conditional independence tests [2]. To reduce the computational cost, other methods provide output which has a sound causal interpretation, but may be less informative. Among these are the anytime FCI [18] and RFCI [2]. A recent algorithm, ancestral causal inference (ACI) [11], aims at learning only the directed part of the underlying graphical structure which allows for a sound causal interpretation even though some information is lost.

In this paper, we describe some simple methods for learning causal structure in dynamical systems represented by stochastic processes. Many authors have described frameworks and algorithms for learning structure in systems of time series, ordinary differential equations, stochastic differential equations, and point processes. However, most of these methods do not have a clear causal interpretation when the observed processes are part of a larger system and most of the current literature is either non-causal in nature, or requires that there are no unobserved processes.

Analogously to testing conditional independence when learning DAGs, one can use tests of local independence in the case of dynamical systems [5, 12, 14]. In [12, 14] the authors propose algorithms for learning local independence structures. We show empirically that we can recover features of the causal structure using considerably fewer tests of local independence. This is done by first suggesting a learning target which is easier to learn, though still conveys useful causal information, analogously to ACI [11]. Second, the proposed algorithm is only guaranteed to provide a supergraph of the learning target and this also reduces the number of local independence tests drastically. A central point is that our proposed methods retain a sound causal interpretation under a faithfulness-type assumption.

In [12], the author suggests learning a directed graph to represent a causal dynamical system and gives a learning algorithm which we will describe as a *simple screening algorithm*. We show that

Preprint. Under review.

this algorithm can be given a sound interpretation under a weaker faithfulness assumption than that of [12]. We also provide a simple interpretation of the output of this algorithm and we show that similar screening algorithms can give comparable results using considerably fewer tests of local independence.

For illustration of the proposed algorithms, we will use linear Hawkes processes in this paper. This model class is used in a wide range of application and is also a topic of methodological research (see e.g. [9] and references therein). All proofs are provided in the supplementary material.

2 Hawkes processes

The algorithmic results we present apply in general to local independence models. To provide a concrete causal model, we will consider the *linear Hawkes processes*. [9] gives an accessible introduction to this model class. On a filtered probability space, $(\Omega, \mathcal{F}, (\mathcal{F}_t), \mathrm{P})$, we consider an n-dimensional multivariate point process, $X = (X^1, \ldots, X^n)$. Each coordinate process X^α is described by a sequence of positive, stochastic event times $T_1^\alpha, T_2^\alpha, \ldots$ such that $T_j^\alpha > T_i^\alpha$ almost surely for $j > i$. We let $V = \{1, \ldots, n\}$. This can also be formulated in terms of a counting process, N, such that $N_s^\alpha = \sum_i \mathbb{1}_{(T_i \leq s)}$, $\alpha \in V$. There exists so-called *intensity processes*, $\lambda = (\lambda^1, \ldots, \lambda^n)$ such that

$$\lambda_t^\alpha = \lim_{h \to 0} \frac{1}{h} \mathrm{P}(N_{t+h}^\alpha - N_t^\alpha = 1 \mid \mathcal{F}_t)$$

and the intensity at time t can therefore be thought of as describing the probability of a jump in the immediate future after time t conditionally on the history until time t as captured by the $\mathcal{F}_t$-filtration. In a linear Hawkes model, the intensity of the α-process, $\alpha \in V$, is of the simple parametric form

$$\lambda_t^\alpha = \mu_\alpha + \sum_{\gamma \in V} \int_0^t g^{\alpha\gamma}(t-s) \, \mathrm{d}N_s^\gamma$$

where $\mu_\alpha \geq 0$ and the functions $g^{\alpha\gamma} : \mathbb{R}_+ \to \mathbb{R}$, $\alpha, \gamma \in V$, are nonnegative. From the above formula, we see that if $g^{\beta\alpha} = 0$, then the α-process does not enter directly into the intensity of the β-process, $\alpha, \beta \in V$ and we will formalize this observation in subsequent sections.

2.1 A dynamical causal model

We will in this section define what is meant by a *causal model* and also define a graph (V, E) which represents the causal structure of the model. The node set V is the index set of the coordinate processes of the multivariate Hawkes process, thus identifying each node with a coordinate process. If we first consider the case where $X = (X_1, \ldots, X_n)$ is a multivariate random variable, it is common to define a *causal* model in terms of a DAG, $\mathcal{D}$, and a structural causal model [15, 16] by assuming that there exists functions f_i and error terms ϵ_i such that

$$X_i = f_i(X_{\mathrm{pa}_\mathcal{D}(X_i)}, \epsilon_i)$$

for $i = 1, \ldots, n$. The causal assumption amounts to assuming that the functional relations are stable under interventions. This idea can be transferred to dynamical systems (see also [17, 14]). If we consider the model described above, we can consider intervening on the α-process and e.g. enforce events in the α-process at the deterministic times $t_1, \ldots, t_k$, and these times only. In this case, the causal assumption amounts to assuming that the distribution of the intervened system is governed by the intensities

$$\lambda_t^\beta = \mu_\beta + \int_0^t g^{\beta\alpha}(t-s) \, \mathrm{d}\bar{N}_s^\alpha + \sum_{\gamma \in V \setminus \{\alpha\}} \int_0^t g^{\beta\gamma}(t-s) \, \mathrm{d}N_s^\gamma$$

for all $\beta \in V \setminus \{\alpha\}$ and where $\bar{N}_t^\alpha = \sum_{i=1}^k \mathbb{1}_{(t \leq t_i)}$. We will not go into a discussion of the existence of these intervened stochastic processes. The above is a *hard* intervention in the sense that the

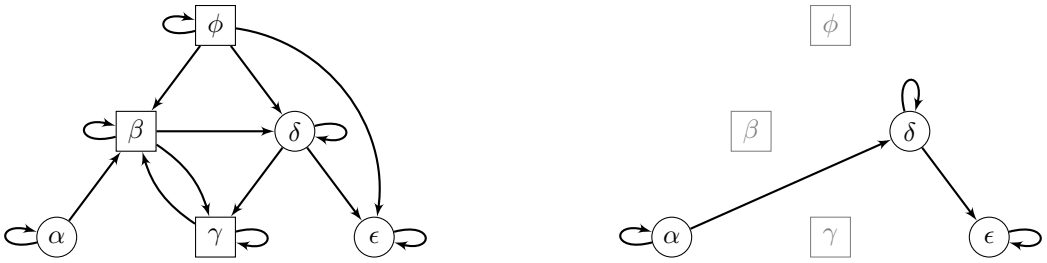


Figure 1: Left: a causal graph on nodes $V = \{\alpha, \beta, \gamma, \delta, \epsilon, \phi\}$. Right: the corresponding parent graph on nodes $O = \{\alpha, \delta, \epsilon\}$. Note that causal graphs and parent graphs may contain cycles. The parent graph does not contain information on the confounder process ϕ as it only encodes 'causal ancestors'.

α-process is fixed to be a deterministic function of time. Note that one could easily imagine other types of interventions such as *soft* interventions where the intervened process is not deterministic. It holds that $N^\beta_{t+h} - N^\beta_t \sim Pois(\lambda^\beta_t \cdot h)$ in the limit $h \to 0$, and we can think of this as a simulation scheme in which we generate the points in one small interval in accordance with some distribution depending on the history of the process. As such the intensity describes a structural causal model at infinitesimal time steps.

We use the set of functions $\{g^{\beta\alpha}\}_{\alpha,\beta\in V}$ to define the *causal graph* of the Hawkes process. A *graph* is a pair (V, E) where V is a set of nodes and E is a set of edges between these nodes. We assume that we observe the Hawkes process in the time interval $J = [0, T], T \in \mathbb{R}$. The causal graph has node set V (the index set of the coordinate processes) and the edge $\alpha \to \beta$ is in the causal graph if and only if $g^{\beta\alpha}$ is not identically zero on J. We call this graph *causal* as it is defined using $\{g^{\beta\alpha}\}_{\alpha,\beta\in V}$ which is a set of mechanisms assumed stable under interventions, and this causal assumption is therefore analogous to that of a classical structural causal model as briefly introduced above.

2.2 Parent graphs

In principle, we would like to recover the causal graph, $\mathcal{D}$, using local independence tests. Often, we will only have partial observation of the dynamical system in the sense that we only observe the processes in $O \subsetneq V$. We will then aim to learn the *parent graph* of $\mathcal{D}$.

Definition 1 (Parent graph)**.** Let $\mathcal{D} = (V, E)$ be a causal graph and let $O \subseteq V$. The *parent graph* of $\mathcal{D}$ on nodes O is the graph (O, F) such that for $\alpha, \beta \in O$, the edge $\alpha \to \beta$ is in F if and only if the edge $\alpha \to \beta$ is in the causal graph or there is a path $\alpha \to \delta_1 \to \ldots \to \delta_k \to \beta$ in the causal graph such that $\delta_1, \ldots, \delta_k \notin O$, for some $k > 0$.

The parent graph thus encodes whether or not an observed process is a 'causal ancestor' of another. We denote the parent graph of the causal graph by $\mathcal{P}_O(\mathcal{G})$, or just $\mathcal{P}(\mathcal{G})$ if the set O used is clear from the context. In applications, a parent graph may provide answers to important questions as it tells us the causal relationships between the observed nodes. A similar idea was applied in [11]. In large systems, it can easily be infeasible to learn the complete independence structure of the observed system, and we propose instead to estimate the parent graph which can be done efficiently. In the supplementary material, we give another characterization of a parent graph. Figure 1 contains an example of a causal graph and a corresponding parent graph.

2.3 Local independence

Local independence has been studied by several authors and in different classes of continuous-time models as well as in time series [1, 3, 4, 6]. We give an abstract definition of local independence, following the exposition in [14].

Definition 2 (Local independence)**.** Let X be a multivariate stochastic process and let V be an index set of its coordinate processes. Let $\mathcal{F}^D_t$ denote the complete and right-continuous version of the σ-algebra $\sigma(\{X^\alpha_s : s \leq t, \alpha \in D\})$. Let λ be a multivariate stochastic process (assumed to be integrable and càdlàg) such that its coordinate processes are indexed by V. For $A, B, C \subseteq V$, we say that X^B is λ-locally independent of X^A given X^C (or simply B is λ-locally independent of A given C) if the process

$$t \mapsto \mathrm{E}(\lambda_t^\beta \mid \mathcal{F}_t^{C \cup A})$$

has an $\mathcal{F}_t^C$-adapted version for all $\beta \in B$. We write this as $A \not\to_\lambda B \mid C$, or simply $A \not\to B \mid C$.

In the case of Hawkes processes, the intensities will be used as the λ-processes in the above definition. See [13, 14] for technical details on the definition of local independence.

2.3.1 Local independence and the causal graph

To make progress on the learning task, we will in this subsection describe the link between the local independence model and the causal graph.

Definition 3 (Pairwise Markov property)**.** We say that a local independence model satisfies the *pairwise Markov property* with respect to a DG, $\mathcal{D} = (V, E)$, if the absence of the edge $\alpha \to \beta$ in $\mathcal{D}$ implies $\alpha \not\to_\lambda \beta \mid V \setminus \alpha$ for all $\alpha, \beta \in V$.

We will make the following technical assumption throughout the paper. In applications, the functions $g^{\alpha\beta}$ are often assumed to be of the below type (see e.g. [9] for common choices of $g^{\beta\alpha}$-functions).

Assumption 4. Assume that N is a multivariate Hawkes process and that we observed N over the interval $J = [0, T]$ where $T > 0$. For all $\alpha, \beta \in V$, the function $g^{\beta\alpha} : \mathbb{R}_+ \to \mathbb{R}$ is continuous on J.

A version of the following result is also stated in [7] but no proof is given. If $\mathcal{G}_1 = (V, E_1)$ and $\mathcal{G}_2 = (V, E_2)$ are graphs, we say that $\mathcal{G}_1$ is a *proper subgraph* of $\mathcal{G}_2$ if $E_1 \subsetneq E_2$.

Proposition 5. The local independence model of a linear Hawkes process satisfies the pairwise Markov property with respect to the causal graph of the process and no proper subgraph of the causal graph has the property.

3 Graph theory and independence models

In order to give full proofs of the statements of the paper, we need a number of graph-theoretical concepts that are otherwise not central to our presentation. We have included these in the supplementary material and in this section we only introduce the most central concepts.

A *graph* is a pair (V, E) where V is a finite set of nodes and E a finite set of edges. An edge can be of different types, and we will use $\sim$ to denote a generic edge of any type. Each edge is between a pair of nodes (not necessarily distinct), and for $\alpha, \beta \in V$, $e \in E$, we will write $\alpha \overset{e}{\sim} \beta$ to denote that the edge e is between α and β. We will in particular consider the class of *directed graphs* (DGs) where between each pair of nodes $\alpha, \beta \in V$ one has a subset of the edges $\{\alpha \to \beta, \alpha \leftarrow \beta\}$, and we say that these edges are *directed*.

Let $\mathcal{G}_1 = (V, E_1)$ and $\mathcal{G}_2 = (V, E_2)$ be graphs. We say that $\mathcal{G}_2$ is a *supergraph* of $\mathcal{G}_1$, and write $\mathcal{G}_1 \subseteq \mathcal{G}_2$, if $E_1 \subseteq E_2$. For a graph $\mathcal{G} = (V, E)$ such that $\alpha, \beta \in V$, we write $\alpha \to_{\mathcal{G}} \beta$ to indicate that the directed edge from α to β is contained in the edge set E. In this case we say that α is a *parent* of β. We let $\mathrm{pa}_{\mathcal{G}}(\beta)$ denote the set of nodes in V that are parents of β. We write $\alpha \not\to_{\mathcal{G}} \beta$ to indicate that the edge is *not* in E. For simplicity, we will throughout the paper assume that all loops, i.e. self-edges $\alpha \to \alpha$, are present in the graphs we consider. This graphical assumption corresponds to implicitly assuming that every process of the dynamical system depends on its own past in a direct fashion.

A *walk* is a finite sequence of nodes, $\alpha_i \in V$, and edges, $e_i \in E$, $\langle \alpha_1, e_1, \alpha_2, \ldots, \alpha_n, e_n, \alpha_{n+1} \rangle$ such that e_i is between α_i and α_{i+1} for all $i = 1, \ldots, n$ and such that an orientation of each edge is known. We say that a walk is *nontrivial* if it contains at least one edge. A *path* is a walk such that no node is repeated. A *directed* path from α to β is a path such that all edges are directed and point in the direction of β.

Definition 6 (Trek, directed trek)**.** A *trek* between α and β is a (nontrivial) path $\langle \alpha, e_1, \ldots, e_n, \beta \rangle$ with no colliders [8]. We say that a trek between α and β is *directed* from α to β if e_n has a head at β. We use $\mathrm{dt}(\beta)$ to denote the set of nodes γ such that there exists a directed trek from γ to β.

As an example, we note that a directed path from α to β is a trek and it is also a directed trek from α to β. However, it is not a directed trek from β to α.

We will formulate the following properties using a general *independence model*, $\mathcal{I}$, on V. Let $\mathbb{P}(\cdot)$ denote the power set of some set. An independence model on V is simply a subset of $\mathbb{P}(V) \times \mathbb{P}(V) \times \mathbb{P}(V)$ and can be thought of as a collection of independence statements that hold among the processes/variables indexed by V. In subsequent sections, the independence models will be defined using the notion of local independence. In this case, for $A, B, C \subseteq V$, $A \not\to_\lambda B \mid C$ is equivalent with writing $\langle A, B \mid C \rangle \in \mathcal{I}$ in the abstract notation, and we use the two interchangeably. In the following, we also use μ-separation which is a ternary relation and a dynamical model (and asymmetric) analogue to d-separation or m-separation. A definition and references are given in the supplementary material. For $\mathcal{G} = (V, E)$ and $A, B, C \subseteq V$ we write $A \perp_\mu B \mid C \, [\mathcal{G}]$ to denote that B is μ-separated from A given C in the graph $\mathcal{G}$.

Definition 7 (Global Markov property)**.** We say that an independence model $\mathcal{I}$ satisfies the *global Markov property* with respect to a DG, $\mathcal{G} = (V, E)$, if $A \perp_\mu B \mid C \, [\mathcal{G}]$ implies $\langle A, B \mid C \rangle \in \mathcal{I}$ for all $A, B, C \subseteq V$.

From Proposition 5, we know that the local independence model of a linear Hawkes process satisfies the pairwise Markov property with respect to the causal graph of the process, and using the results in [4, 14] it also satisfies the global Markov property with respect to this graph.

Definition 8 (Faithfulness)**.** We say that $\mathcal{I}$ is *faithful* with respect to a DG, $\mathcal{G} = (V, E)$, if $\langle A, B \mid C \rangle \in \mathcal{I}$ implies $A \perp_\mu B \mid C \, [\mathcal{G}]$ for all $A, B, C \subseteq V$.

4 Learning algorithms

In this section, we will define a faithfulness-type assumption that will be used to ensure the soundness of the learning algorithms. We then state a very general class of algorithms which is easily seen to provide sound causal learning and we describe some specific algorithms.

We throughout assume that there is some underlying DG, $\mathcal{D}_0 = (V, E)$, describing the causal model and we wish to output $\mathcal{P}_O(\mathcal{D}_0)$. However, this graph is not in general identifiable from the local independence model. In the supplementary material, we argue that for an equivalence class of parent graphs, there exists a unique member of the class which is a supergraph of all other members. Denote this unique graph by $\bar{\mathcal{D}}$. Our algorithms will output supergraphs of $\bar{\mathcal{D}}$, and the output will therefore also be supergraphs of the true parent graph.

We assume we are in the 'oracle case', i.e. have access to a local independence oracle that provides the correct answers. We will say that an algorithm is *sound* if it in the oracle case outputs a supergraph of $\bar{\mathcal{D}}$ and that it is *complete* if it outputs $\bar{\mathcal{D}}$. We let $\mathcal{I}^O$ denote the local independence model restricted to subsets of O, i.e. this is observed part of the local independence model.

4.1 Ancestral faithfulness

Under the faithfulness assumption, every local independence implies μ-separation in the graph. We assume a weaker, but similar, property to argue that the our algorithms are sound. For learning marginalized DAGs, weaker types of faithfulness have also been explored, see e.g. [24, 21, 22].

Definition 9 (Ancestral faithfulness)**.** Let $\mathcal{I}$ be an independence model and let $\mathcal{D}$ be a DG. We say that $\mathcal{I}$ satisfies *ancestral faithfulness* with respect to $\mathcal{D}$ if for every $\alpha, \beta \in V$ and $C \subseteq V \setminus \{\alpha\}$, $\langle \alpha, \beta \mid C \rangle \in \mathcal{I}$ implies that there is no μ-connecting directed path from α to β given C in $\mathcal{D}$.

It follows from the definition that faithfulness implies ancestral faithfulness and for general independence models ancestral faithfulness is a strictly weaker property than faithfulness. We conjecture that local independence models of linear Hawkes processes satisfy ancestral faithfulness with respect to their causal graphs. Heuristically, if there is a directed path from α to β which is not blocked by any node in C, then information should flow from α to β, and this cannot be 'cancelled out' by other paths in the graph as the linear Hawkes processes are self-excitatory, i.e. no process has a dampening effect on any process.

4.2 Simple screening algorithms

As a first step in describing a causal screening algorithm, we will define a very general class of learning algorithms that simply test local independences and sequentially remove edges. It is easily seen that

under the assumption of ancestral faithfulness every algorithm in this class gives sound learning in the oracle case. The *complete* DG on nodes V is the DG with edge set $\{\alpha \to \beta \mid \alpha, \beta \in V\}$.

Definition 10 (Simple screening algorithm)**.** We say that a learning algorithm is a *simple screening algorithm* if it starts from a complete DG on nodes O and removes an edge $\alpha \to \beta$ only if a conditioning set $C \subseteq O \setminus \{\alpha\}$ has been found such that $\langle \alpha, \beta \mid C \rangle \in \mathcal{I}^O$.

The following results give a clear and causally sound interpretation of the output of a simple screening algorithm.

Proposition 11. Assume that $\mathcal{I}$ satisfies ancestral faithfulness with respect to $\mathcal{D}_0 = (V, E)$. The output of any simple screening algorithm is sound in the oracle case.

Corollary 12. Assume ancestral faithfulness of $\mathcal{I}$ with respect to $\mathcal{D}_0$ and let $A, B, C \subseteq O$. If every directed path from A to B goes through C in the output graph of a simple screening algorithm, then every directed path from A to B goes through C in $\mathcal{D}_0$.

Corollary 13. If there is no directed path from A to B in the output graph, then there is no directed path from A to B in $\mathcal{D}_0$.

4.3 Parent learning

In the previous section, it was shown that if edges are only removed when a separating set is found the output is sound under the assumption of ancestral faithfulness. In this section we give a specific algorithm. The key observation is that we can easily retrieve structural information from a rather small subset of local independence tests.

Let $\mathcal{D}^t$ denote the output from Subalgorithm 1 (see below). The following result shows that under the assumption of faithfulness, $\alpha \to_{\mathcal{D}^t} \beta$ if and only if there is a directed trek from α to β in $\mathcal{D}_0$.

Proposition 14. There is no directed trek from α to β in $\mathcal{D}_0$ if and only if $\alpha \perp_\mu \beta \mid \beta\ [\mathcal{D}_0]$.

We will refer to running first Subalgorithm 1 and then Subalgorithm 2 (using the the output DG from the first as input to the second) as the causal screening (CS) algorithm. The following proposition follows directly from the definitions of the subalgorithms.

Proposition 15. The CS algorithm is a simple screening algorithm.

It is of course of interest to understand under what conditions the edge $\alpha \to \beta$ is guaranteed to be removed by the CS algorithm when it is not in the underlying target graph. In the supplementary material we state and prove a result describing one such condition.

input : a local independence oracle for $\mathcal{I}^O$
output : a DG on nodes O
initialize $\mathcal{D}$ as the complete DG on O;
foreach $(\alpha, \beta) \in V \times V$ **do**
  if $\alpha \not\to_\lambda \beta \mid \beta$ **then**
    delete $\alpha \to \beta$ from $\mathcal{D}$;
  end
end
return $\mathcal{D}$

Subalgorithm 1: Trek step

input : a local independence oracle for $\mathcal{I}^O$ and a DG, $\mathcal{D} = (O, E)$
output : a DG on nodes O
foreach $(\alpha, \beta) \in V \times V$ *such that* $\alpha \to_{\mathcal{D}} \beta$ **do**
  if $\alpha \not\to_\lambda \beta \mid pa_{\mathcal{D}}(\beta) \setminus \{\alpha\}$ **then**
    delete $\alpha \to \beta$ from $\mathcal{D}$;
  end
end
return $\mathcal{D}$

Subalgorithm 2: Parent step

4.4 Ancestry propagation

In this section, we describe an additional step which propagates ancestry by reusing the output of Subalgorithm 1 to remove further edges. This comes at a price as one needs to assume faithfulness in order to guarantee that the result will be sound. The idea is similar to ACI [11] that also uses that ancestry is transitive.

In ancestry propagation, we exploit the fact that any trek between α and β composed with the edge $\beta \to \gamma$ gives a directed trek from α to β. We only use the trek between α and β 'in one direction' and this is because we should be slightly careful if γ is actually on the trek between α and β. In

input : a local independence oracle for $\mathcal{I}^O$ and a DG, $\mathcal{D} = (O, E)$
output : a DG on nodes O
initialize $E_r = \emptyset$ as the empty edge set;
foreach $(\alpha, \beta, \gamma) \in V \times V \times V$ *such that* α, β, γ *are all distinct* **do**
  if $\alpha \to_{\mathcal{D}} \beta$, $\beta \not\to_{\mathcal{D}} \alpha$, $\beta \to_{\mathcal{D}} \gamma$, *and* $\alpha \not\to_{\mathcal{D}} \gamma$ **then**
    update $E_r = E_r \cup \{\beta \to \gamma\}$;
  end
end
Update $\mathcal{D} = (V, E \setminus E_r)$;
return $\mathcal{D}$

Subalgorithm 3: Ancestry propagation

Subalgorithm 4 in the supplementary material, we exploit a trek between α and β twice, but at the cost of an additional local independence test.

We can construct an algorithm by first running Subalgorithm 1, then Subalgorithm 3, and finally Subalgorithm 2 (using the output of one subalgorithm as input to the next). We will call this the CSAPC algorithm. If we use Subalgorithm 4 instead of Subalgorithm 3, we will call this the CSAP.

Proposition 16. If $\mathcal{I}$ is faithful with respect to $\mathcal{D}_0$, then the CSAP and CSAPC algorithms both provide sound learning.

5 Application and simulations

When evaluating the performance of a sound screening algoritm, the output graph is guaranteed to be a supergraph of the true parent graph, and we will say that edges that are in the output but not in the true graph are *excess edges*. For a node in a directed graph, the *indegree* is the number of directed edges adjacent with and pointed into the node, and the *outdegree* is the number of directed edges adjacent with and pointed away from the node.

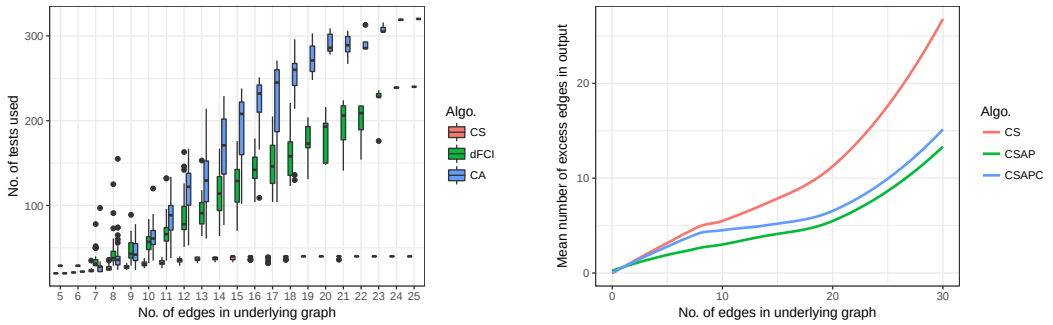
5.1 C. elegans neuronal network

Caenorhabditis elegans is a roundworm in which the network between neurons has been mapped completely [20]. We apply our methods to this network as an application to a highly complex network. It consists of 279 neurons which are connected by both non-directional *gap junctions* and directional chemical synapses. We will represent the former as an unobserved process and the latter as a direct influence. From this network, we sampled subnetworks of 75 neurons each (details in the supplementary material) and computed the output of the CS algorithm. These subsampled networks had on average 1109 edges (including bidirected edges representing unobserved processes, see the supplementary material) and on average 424 directed edges. The output graphs had on average 438 excess edges which is explained by the fact that there are many unobserved nodes in the graphs. To compare the output to the true parent graph, we computed the rank correlation between the indegrees of the nodes in the output graph and the indegrees of the nodes in the true parent graph, and similarly for the outdegree (indegree correlation: 0.94, outdegree correlation: 0.52). Finally, we investigated the method's ability to identify the observed nodes of highest directed connectivity (i.e. highest in- and outdegrees). The neuronal network of c. elegans is inhomogeneous in the sense that some neurons are extremely highly connected while others are only very sparsely connected. Identifying such highly connected neurons is of interest, and we considered the 15 nodes of highest indegree/outdegree (out of the 75 observed nodes). On average, the CS algorithm placed 13.4 (in) and 9.2 (out) of these 15 among the 15 most connected nodes in the output graph.

From the output of the CS algorithm, we can easily find areas of the neuronal network which mediates the information from one area to another, e.g. using Corollary 12.

5.2 Comparison of algorithms

In this section we compare the proposed causal screening algorithms with previously published algorithms that solve similar problems. In [14], the authors propose two algorithms, one of which is



(a) Comparison of number of tests used between CS, dFCI, and CA. We generated 500 graphs, all on 5 nodes. Mean number of excess edges: CS 0.96, dFCI 0.07, CA 0.81.

(b) Mean number of excess edges in output graphs for varying numbers of edges (bidirected and directed) in the true graph (all graphs are on 10 nodes) not counting loops.

Figure 2: Comparison of performance.

sure to output the correct graph. The authors note that this complete algorithm is computationally very expensive and adds little extra information, and therefore we will only consider their other algorithm for comparison. We will call this algorithm *dynamical* FCI (dFCI) as it resembles the FCI algorithm as noted by the authors of [14]. The algorithm actually solves a harder learning problem and provides more information (see details in the supplementary material), however it is computationally infeasible for many problems.

The Causal Analysis (CA) algorithm of [12] is a simple screening algorithm and we have in this paper argued that it is sound for learning the parent graph of the underlying graph under the weaker assumption of ancestral faithfulness. Note that even though this algorithm uses a large number of tests, it is not guaranteed to provide complete learning as there may be inseparable nodes that are not adjacent [13, 14].

For the comparison of these algorithms, two aspects are important. As they are all sound, one aspect is the number of excess edges. The other aspect is of course the number of tests needed to provide their output. The CS and CSAPC algorithms use at most $2n(n-1)$ tests and empirically the CSAP uses roughly the same number as the two former. This makes them feasible in even large graphs. The quality of their output is dependent on the sparsity of the graph, though the CSAP and CSAPC algorithms can deal considerably better with less sparse graphs (Subfigure 2b).

6 Discussion

We suggested inexpensive constraint-based methods for learning causal structure based on testing local independence. An important observation is that local independence is asymmetric while conditional independence is symmetric. In a certain sense, this may help when constructing learning algorithms as there is no need of something like an 'orientation phase' as in the FCI. This facilitates using very simple methods to give sound causal learning as we do not need the independence structure in full to give interesting output.

The level of information in the output graph of the causal screening algorithms is dependent on the sparsity of the graph. However, even in examples with very little sparsity, as in the c.elegans neuronal network, interesting structural information can be learned using these simple methods.

By outputting only the directed part of the underlying causal structure, we may be able to answer structural questions, but not other questions e.g. relating to causal effect estimation. However, by restricting the scope we can provide a sound algorithm which can reveal interesting information about the causal structure.

We showed that the proposed algorithms have a large computational advantage over previously published algorithms within this framework. This makes it feasible to consider causal learning even in large networks with unobserved processes and the CS algorithms thus provide methods for sound screening in causal dynamical models.

References

[1] Odd O. Aalen. Dynamic modelling and causality. *Scandinavian Actuarial Journal*, pages 177–190, 1987.

[2] Diego Colombo, Marloes H. Maathuis, Markus Kalisch, and Thomas S. Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1):294–321, 2012.

[3] Vanessa Didelez. *Graphical Models for Event History Analysis based on Local Independence*. PhD thesis, Universität Dortmund, 2000.

[4] Vanessa Didelez. Graphical models for marked point processes based on local independence. *Journal of the Royal Statistical Society, Series B*, 70(1):245–264, 2008.

[5] Michael Eichler. Causal inference with multiple time series: principles and problems. *Philosophical Transactions of the Royal Society*, 371(1997):1–17, 2013.

[6] Michael Eichler and Vanessa Didelez. On Granger causality and the effect of interventions in time series. *Lifetime Data Analysis*, 16(1):3–32, 2010.

[7] Michael Eichler, Rainer Dahlhaus, and Johannes Dueck. Graphical modeling for multivariate Hawkes processes with nonparametric link functions. *Journal of Time Series Analysis*, 38: 225–242, 2017.

[8] Rina Foygel, Jan Draisma, and Mathias Drton. Half-trek criterion for generic identifiability of linear structural equation models. *The Annals of Statistics*, 40(3):1682–1713, 2012.

[9] Patrick J. Laub, Thomas Taimre, and Philip K. Pollett. Hawkes processes. 2015. URL `https://arxiv.org/pdf/1507.02822.pdf`.

[10] Marloes Maathuis, Mathias Drton, Steffen Lauritzen, and Martin Wainwright. *Handbook of graphical models*. Chapman & Hall/CRC handbooks of modern statistical methods, 2019.

[11] Sara Magliacane, Tom Claassen, and Joris M. Mooij. Ancestral causal inference. In *Proceedings of the 29th Conference on Neural Information Processing Systems (NIPS 2016)*, 2016.

[12] Christopher Meek. Toward learning graphical and causal process models. In *CI'14 Proceedings of the UAI 2014 Conference on Causal Inference: Learning and Prediction*, 2014.

[13] Søren Wengel Mogensen and Niels Richard Hansen. Markov equivalence of marginalized local independence graphs. 2019. To appear in the Annals of Statistics.

[14] Søren Wengel Mogensen, Daniel Malinsky, and Niels Richard Hansen. Causal learning for partially observed stochastic dynamical systems. In *Proceedings of the 34th conference on Uncertainty in Artificial Intelligence*, 2018.

[15] Judea Pearl. *Causality*. Cambridge University Press, 2009.

[16] Jonas Christopher Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference, foundations and learning algorithms*. MIT Press, 2017.

[17] Kjetil Røysland. Counterfactual analyses with graphical models based on local independence. *The Annals of Statistics*, 40(4):2162–2194, 2012.

[18] Peter Spirtes. An anytime algorithm for causal inference. In *Proceedings of the Eighth International Workshop on Artificial Intelligence and Statistics, AISTATS 2001*, 2001.

[19] Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT Press, 2000.

[20] Lav R. Varshney, Beth L. Chen, Eric Paniagua, David H. Hall, and Dmitri B. Chklovskii. Structural properties of the Caenorhabditis elegans neuronal network. *PLoS Computational Biology*, 7(2), 2011.

[21] Zhalama, Jiji Zhang, Frederick Eberhardt, and Wolfgang Mayer. SAT-based causal discovery under weaker assumptions. In *Proceedings of the 33th Conference on Uncertainty in Artificial Intelligence*, 2017.

[22] Zhalama, Jiji Zhang, and Wolfgang Mayer. Weakening faithfulness: some heuristic causal discovery algorithms. *International Journal of Data Science and Analytics*, 3:93–104, 2017.

[23] Jiji Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172:1873–1896, 2008.

[24] Jiji Zhang and Peter Spirtes. Detection of unfaithfulness and robust causal inference. *Minds & Machines*, 18:239–271, 2008.