On the Definition and Computation of Causal Treewidth (Supplementary material)

Yizuo Chen[1] **Adnan Darwiche**[1]

[1]Computer Science Department, University of California, Los Angeles, USA

1 PROOFS

Our proofs are ordered sightly differently than the corresponding results in the paper as we need some results when proving others. In the upcoming proofs, we will also use f_X to denote a mechanism for variable X. We will also say that a leaf node i in a jointree *contains* variable X iff X appears in a factor that is hosted at leaf node i.

1.1 PROOF OF THEOREM 5

See [Darwiche, 2009, Ch 7].

1.2 PROOF OF THEOREM 6

Follows directly from Definitions 9 and 10.

1.3 PROOF OF THEOREM 10

In this proof, we will assume we have a mapping from jointree edges (i, j) to sets $\mathbf{S}'_{ij} \subseteq \mathbf{S}_{ij}$ that satisfy conditions (a,b,c) of Definition 8 (that is, these sets satisfy only the first part of this definition but not the second part so they are not necessarily thinned separators). We will also say that jointree nodes i and j are $(X, \mathbf{S}')$-connected iff $i = j$ or variable X appears in every set $\mathbf{S}'$ that is attached to an edge on the path between i and j. We will first state and prove two lemmas which we need for the proof of this theorem.

Lemma 1. *If $X \in \mathbf{S}'_{ij}$, then node i is $(X, \mathbf{S}')$-connected to some leaf node on the i-side of edge (i, j) which contains variable X and node j is $(X, \mathbf{S}')$-connected to some leaf node on the j-side of the edge which also contains X.*

Proof. Suppose node i is not $(X, \mathbf{S}')$-connected to some leaf node on the i-side of edge (i, j) which contains X. Consider a longest path $i, \ldots, r, l$ on the i-side of the edge such that X appears in $\mathbf{S}'$ for each edge on the path. If l is a leaf node, then it must contain variable X by definition of separators, which is a contradiction. Suppose now that l is not a leaf node. Then $X \notin \mathbf{S}'_{lk}$ for $k \neq r$, otherwise the path would not be longest. This contradicts the assumed condition (b) of Definition 8 on sets $\mathbf{S}'$. Hence, node i must be $(X, \mathbf{S}')$-connected to some leaf node that contains X on the i-side of edge (i, j). We can similarly show the second part of the lemma. $\square$

Lemma 2. *Suppose sets $\mathbf{S}'_{ij}$ were obtained by exhausting thinning rules on separators $\mathbf{S}_{ij}$. If node k is $(X, \mathbf{S}')$-connected to some leaf node that contains X, then node k is $(X, \mathbf{S}')$-connected to exactly one leaf node that hosts f_X.*

Proof. Suppose node k is $(X, \mathbf{S}')$-connected to some leaf node that contains X. Then node k cannot be $(X, \mathbf{S}')$-connected to more than one leaf node hosting f_X; otherwise, Rule (a) will apply. We next show that node k must be $(X, \mathbf{S}')$-connected to at least one mechanism f_X. We will show this by induction on the number of rule applications. We will use $\mathbf{S}^n$ to denote the state of separators after the n^{th} rule application. For the base case (before any thinning rules are applied), node k must be (X, S^0)-connected to some f_X by the definition of separators in a jointree. For the inductive step, suppose node k is $(X, \mathbf{S}^n)$-connected to some leaf node that contains X only if node k is $(X, \mathbf{S}^n)$-connected to at least one leaf node that hosts f_X. Suppose now that node k is $(X, \mathbf{S}^{n+1})$-connected to some leaf node that contains X. We will next show that node k must be $(X, \mathbf{S}^{n+1})$-connected to at least one mechanism f_X.

First, node k must be $(X, \mathbf{S}^n)$-connected to some leaf node that contains X. Hence, by the induction hypothesis, node k must be $(X, \mathbf{S}^n)$-connected to at least one mechanism f_X. We will consider an edge (i, j) such that $X \in \mathbf{S}^n_{ij}$ and $X \notin \mathbf{S}^{n+1}_{ij}$ (such an edge must exist) and do a case analysis on which rule applied to this edge.

Case: Rule (a). We will show next that node k must be $(X, \mathbf{S}^{n+1})$-connected to at least one mechanism f_X while assuming that node k is on the i-side of edge (i, j). A similar argument will show the same if node k is on the j-side of

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

the edge. Suppose node k is on the i-side of edge (i, j). If node k is $(X, \mathbf{S}^n)$-connected to some f_X on the i-side, then node k is $(X, \mathbf{S}^{n+1})$-connected to the same f_X on the i-side. If node k is $(X, \mathbf{S}^n)$-connected to some f_X on the j-side, then node k is $(X, \mathbf{S}^{n+1})$-connected to node i. By the definition of Rule (a), node i must be $(X, \mathbf{S}^{n+1})$-connected to some f_X on the i-side. Therefore, node k must be $(X, \mathbf{S}^{n+1})$-connected to the same f_X on the i-side.

Case: Rule (b). Then none of the neighboring separators except $\mathbf{S}^n_{ij}$ contains X. If node k is on the i-side, then node k must be $(X, \mathbf{S}^n)$ to some f_X on the i-side and therefore must be $(X, \mathbf{S}^{n+1})$ to the same f_X on the i-side. The same argument applies if node k is on the j-side.

Case: Rule (c). Symmetric to the previous case. □

Proof of Theorem 10. First note that conditions (a,b,c) of Definition 8 hold when we exhaust thinning rules. Suppose sets $\mathbf{S}'_{ij}$ were obtained by exhausting thinning rules on separators $\mathbf{S}_{ij}$ (and hence satisfy the three conditions). We just need to prove that no supersets of $\mathbf{S}'_{ij}$ satisfy these conditions. Suppose by contradiction such supersets $\mathbf{S}''_{ij}$ exist and consider an edge (i, j) such that $X \in \mathbf{S}''_{ij}$ and $X \notin \mathbf{S}'_{ij}$. By Lemma 1 since sets $\mathbf{S}''_{ij}$ satisfy the three conditions, node i must $(X, \mathbf{S}'')$-connect to some leaf node k on the i-side which contains X. Since leaf node k contains X, it $(X, \mathbf{S}')$-connects to itself. By Lemma 2, the leaf node k is $(X, \mathbf{S}')$-connected to some f_X on the i-side since $X \notin \mathbf{S}'_{ij}$. Hence, leaf node k is also $(X, \mathbf{S}'')$-connected to the same f_X on the i-side since sets $\mathbf{S}''_{ij}$ are supersets of $\mathbf{S}'_{ij}$. We have shown that node i is $(X, \mathbf{S}'')$-connected to leaf node k which is $(X, \mathbf{S}'')$-connected to some f_X on the i-side, therefore node i is $(X, \mathbf{S}'')$-connected to some f_X on the i-side. By a similar argument, node j is $(X, \mathbf{S}'')$-connected to some f_X on the j-side. However, this implies that the supersets $\mathbf{S}''_{ij}$ violate condition (a), which is a contradiction. Hence, the supersets $\mathbf{S}''_{ij}$ cannot exist. □

1.4 PROOF OF THEOREM 11

We need the following lemma which states the same property of Lemma 2 except under different conditions. The proof of this lemma uses the notion of “a closest leaf k to node i which hosts mechanism f_X." This is a leaf node that hosts f_X where the path $i = p_0, p_1, \ldots, p_{n-1}, p_n = k$ has a minimal number of sets $\mathbf{S}^\star_{p_i p_{i+1}}$ that do not contain X.

Lemma 3. *Consider thinned separators $\mathbf{S}^\star$ according to Definition 8 and let X be a functional variable. If node i is $(X, \mathbf{S}^\star)$-connected to some leaf node that contains X, then node i is $(X, \mathbf{S}^\star)$-connected to exactly one leaf node that hosts mechanism f_X.*

Proof. Suppose node i is $(X, \mathbf{S}^\star)$-connected to some leaf node that contains X. Node i cannot be $(X, \mathbf{S}^\star)$-connected to two different leaves that host f_X as this would violate condition (a) of Definition 8. Suppose now that node i is not $(X, \mathbf{S}^\star)$-connected to any mechanism f_X. We will next show a contradiction. Consider the path $i = p_0, p_1, \ldots, p_{n-1}, p_n = k$ where k is a closest leaf to node i which hosts mechanism f_X. We claim that adding X to all sets $\mathbf{S}^\star_{p_i p_{i+1}}$ which do not contain X on the path results in a jointree thinning that still satisfies conditions (a,b,c) of Definition 8. This would be a contradiction as it implies there is a superset of thinning $\mathbf{S}^\star$ that satisfies these conditions.

To show the above claim, note that conditions (b,c) will immediately continue to be satisfied if we add variables to sets $\mathbf{S}^\star_{p_i p_{i+1}}$. We next show that condition (a) will continue to be satisfied as well.

Suppose we are adding X to sets $\mathbf{S}^\star_{p_i p_{i+1}}$ which do not contain variable X using the given order of these sets. Let $\mathbf{S}'$ be the separators after adding X to $\mathbf{S}^\star_{p_i p_{i+1}}$ when the first violation to condition (a) takes place. Then node p_i will be $(X, \mathbf{S}')$-connected to a leaf node p'_i that hosts mechanism f_X on the p_i-side of edge (p_i, p_{i+1}). This contradicts with the definition of k as p'_i will be closer to i than k. □

Proof of Theorem 11. Given a jointree thinning $\mathbf{S}^\star$ that satisfies Definition 8, we will next show how to construct a thinning sequence that produces it.

A key observation is that rule applications for a variable X are independent of rule applications for a variable $Y \neq X$; that is, we can always rearrange a thinning sequence so rules that apply to the same variable are consecutive. Hence, we will construct a thinning sequence that produces $\mathbf{S}^\star$ by constructing a set of rule applications for each variable and then paste them together.

To construct the rule applications for variable X, we start with some leaf node l which hosts mechanism f_X and then traverse nodes away from l. Suppose we are visiting node i now which has a neighbor j that has not been visited. If $X \notin \mathbf{S}^\star_{ij}$ and $X \in \mathbf{S}_{ij}$, we consider two cases. If $X \notin \mathbf{S}^\star_{jk}$ for all k, we add $R_c(i, j, X)$. Otherwise, we add $R_a(i, j, X)$. Every variable that has been thinned will now be accounted for by a rule. Moreover, these rules will be applicable in the reverse order in which they have been constructed. Each $R_a(i, j, X)$ will be applicable by Lemmas 1 and 3. Each $R_c(i, j, X)$ will be applicable by definition. □

1.5 PROOF OF THEOREM 9

Proof. A jointree thinning according to Definition 8 satisfies the two properties of the theorem by Definition 8 and Lemma 3. Suppose now the two properties hold for a mapping $\mathbf{S}^\star$. We will show there exists a thinning sequence that produces $\mathbf{S}^\star$ from the classical separators $\mathbf{S}$. We first note that no thinning rules can be applied to a mapping $\mathbf{S}^\star$ that satisfies the two properties of the theorem. To apply thinning rules to separators $\mathbf{S}$, we consider each functional variable X, then locate all connected subtrees where X does not appear in any separator of a subtree. Consider one such connected subtree Γ and let $\mathcal{B} = \{\mathbf{S}^\star_{u_1t_1}, \ldots, \mathbf{S}^\star_{u_kt_k}\}$ be the boundary separators of the subtree. WLG, assume u_i are at the subtree boundary. Observe that each u_i can either be a leaf node that hosts a mechanism f_X or a non-leaf node such that $X \in \mathbf{S}^\star_{u_ir_i}$ for $r_i \neq t_i$. Similar to the proof for Lemma 1, we can show that each u_i is $(X, \mathbf{S}^\star)$-connected to some f_X. Let $\{\mathbf{S}_{u_1t_1}, \ldots, \mathbf{S}_{u_kt_k}\}$ be the original classical separators of the jointree, then the separators $\mathbf{S}^\star$ for the connected subtree Γ can be obtained by the following thinning sequence. We first apply thinning Rule (a) to all boundary separators $\mathbf{S}_{u_1t_1}, \ldots, \mathbf{S}_{u_kt_k}$ but one. Suppose we apply the rule to separators $\mathbf{S}_{u_2t_2} \ldots, \mathbf{S}_{u_kt_k}$. This is sound since u_1 is $(X, \mathbf{S})$-connected to all $u_2, \ldots, u_k$ by the property of classical separators. Starting from these thinned boundary separators, we can then thin X from all separators in subtree Γ using Rules (b,c). $\square$

1.6 PROOF OF THEOREM 8

Proof. Suppose $\mathbf{S}^\star$ is a jointree thinning according to Definition 8 and and let X be a functional variable such that $X \in \mathbf{S}^\star_{ij}$ for some edge (i, j). By Theorem 9, we must have some leaf node k with $X \in \texttt{vars}(i)$ that is X-connected to exactly one mechanism for X, f_X, through edge (i, j). WLG, suppose leaf k is on the i-side of the edge and mechanism f_X is on the j-side of the edge. Suppose further that we remove variable X from $\mathbf{S}^\star_{ij}$ leading to new separators $\mathbf{S}'$. This will lead to a violation of Condition (2a) in Theorem 9. In particular, leaf node k will no longer be X-connected to any mechanism for X. Now let $\mathcal{F}$ be the factors on the i-side of edge (i, j), $\mathcal{G}$ be the factors on the j-side of the edge and m be the number of mechanisms f_X in $\mathcal{F} \cup \mathcal{G}$. Let $\mathcal{M}^\star$ denote messages computed using separators $\mathbf{S}^\star$ and $\mathcal{M}'$ denote the messages computed using the separators $\mathbf{S}'$. The computation of messages $\mathcal{M}^\star_{ij}$ and $\mathcal{M}^\star_{ji}$ must involve at least $m-1$ distinct sum-outs of X. This follows because the mechanisms f_X cannot be X-connected. Since $\mathcal{M}'_{ij} = \sum_X \mathcal{M}^\star_{ij}$ and $\mathcal{M}'_{ji} = \sum_X \mathcal{M}^\star_{ji}$, computing the product $\mathcal{M}'_{ij} \cdot \mathcal{M}'_{ji}$ involves $(m-1)+2 = m+1$ distinct sum-outs of variable X. If $\Pr(\mathbf{S}'_{ij}) = \sum^{=}_{\mathbf{S}'_{ij}} \mathcal{F} \cdot \mathcal{G} = \mathcal{M}'_{ij} \cdot \mathcal{M}'_{ji}$ then $\mathcal{M}'_{ij} \cdot \mathcal{M}'_{ji}$ is a factorization of $\sum^{=}_{\mathbf{S}'_{ij}} \mathcal{F} \cdot \mathcal{G}$ that involves $m+1$ distinct sum-outs of variable X. However, since factors $\mathcal{F} \cup \mathcal{G}$ contain exactly m replicas of mechanism f_X, any factorization of $\sum^{=}_{\mathbf{S}'_{ij}} \mathcal{F} \cdot \mathcal{G}$ cannot include more than m distinct sum-outs of variable X that are based on Theorems 2 and 3. This follows because each sum-out of X based on Theorem 3 will consume a mechanism f_X and the sum-outs based on Theorem 2 do not consume mechanisms. Hence, the equality $\sum^{=}_{\mathbf{S}'_{ij}} \mathcal{F} \cdot \mathcal{G} = \mathcal{M}'_{ij} \cdot \mathcal{M}'_{ji}$ cannot be justified based only on these two theorems. That is, Theorem 3 will no longer be sufficient to imply the soundness of the message-passing algorithm as stated in Theorem 5. $\square$

1.7 PROOF OF THEOREM 12

Lemma 4. *Consider a thinning sequence* $\mathbf{R} = \{\mathbf{X}, R_k, \mathbf{Y}, R_t, \mathbf{Z}\}$ *where* $\mathbf{X} = \{R_1, \ldots, R_{k-1}\}$, $\mathbf{Y} = \{R_{k+1}, \ldots, R_{t-1}\}$ *and* $\mathbf{Z} = \{R_{t+1}, \ldots, R_n\}$. *Suppose* R_k *and* R_t *are applications of Rule (a) and no member of* $\mathbf{Y}$ *is an application of Rule (a). Then the following is a valid thinning sequence* $\mathbf{R}' = \{\mathbf{X}, R_k, R_t, \mathbf{Y}, \mathbf{Z}\}$.

Proof. When applying the thinning sequence $\mathbf{R}$, we start with $\mathbf{S}^\star_{ij} = \mathbf{S}_{ij}$ and reduce a set $\mathbf{S}^\star_{ij}$ after each rule application. The key observation here is that if an application of Rule (a) is valid at some state of the thinned separators, it will be valid at any earlier state of these separators (because no thinned separator can be smaller at an earlier state). Moreover, if an application of Rules (b,c) is valid at some state of the thinned separators, it will be valid at any later state of these separators. $\square$

Proof of Theorem 12. Consider a thinning sequence $\mathbf{R}$. We can apply Lemma 4 repeatedly to obtain a valid thinning sequence $\mathbf{R}'$ that has the same rule applications as $\mathbf{R}$ and in which Rules (a) appear before Rules (b,c). The sequences $\mathbf{R}$ and $\mathbf{R}'$ generate the same jointree thinning since they contain the exact same rule applications. $\square$

1.8 PROOF OF THEOREM 7

Our proof starts with the correctness of the message passing algorithm using classical separators and then shows that the algorithm continues to be sound after we apply a thinning rule to remove a variable from some separator (recall that every jointree thinning can be obtained by a sequence of thinning rules). We will use $\mathbf{S}'$ to indicate the state of separators after some rule applications.

Our proof uses the following variant on Definition 7. We will say that a jointree node i is *strongly* $(X, \mathbf{S}')$-connected to a factor f iff i hosts f or $\texttt{vars}(f)$ appears in every separator $\mathbf{S}'_{kl}$ on the path between node i and some leaf node j that hosts f. Similarly, we will say that jointree nodes i and j are *strongly* $(X, \mathbf{S}')$-connected iff $i = j$ or variables $\texttt{vars}(f_X)$ appear in every separator $\mathbf{S}'_{kl}$ on the path between i and j.

The thinning rules for distinct variables do not interact with one another. Hence, we will assume in this proof that all thinning rules are applied according to a reverse topological ordering π of the variables in the underlying DAG.

We will use two lemmas in this proof. The first says that X-connection (Definition 7) and strong X-connection (defined above) are equivalent when applying thinning rules according to reverse topological ordering π.

Lemma 5. *Suppose $\mathbf{S}'$ is the state of separators after applying thinning rules to variables that do not follow variable X in order π. For jointree edge (i,j), node i is $(X,\mathbf{S}')$-connected to some f_X on the i-side of the edge and node j is $(X,\mathbf{S}')$-connected to some f_X on the j-side of the edge iff i is strongly $(X,\mathbf{S}')$-connected to some f_X on the i-side and j is strongly $(X,\mathbf{S}')$-connected to some f_X on the j-side.*

Proof. The if part follows from the fact that $X \in \mathtt{vars}(f_X)$. We next show the only-if part. Suppose node i is $(X,\mathbf{S}')$-connected to node l that hosts f_X on the i-side and node j is $(X,\mathbf{S}')$-connected to node r that hosts f_X on the j-side. By the property of classical separators, $\mathtt{vars}(f_X) \subseteq \mathbf{S}_{xy}$ for all edges (x,y) on the path between l and r. Since π is a reverse topological ordering of the variables, none of the parents of X ($\mathtt{vars}(f_X) \setminus \{X\}$) are thinned from the separators. Therefore, $\mathtt{vars}(f_X) \setminus \{X\} \subseteq \mathbf{S}'_{xy}$ for all edges (x,y) on the path between l and r. Since node i is $(X,\mathbf{S}')$-connected to l and node j is $(X,\mathbf{S}')$-connected to r, we conclude that node i is strongly $(X,\mathbf{S}')$-connected to l and node j is strongly $(X,\mathbf{S}')$-connected to r. $\square$

The second lemma extends Theorem 3 to a more general setting. For factors f and $\mathcal{F}$, we will write $f \in \mathcal{F}$ to mean that $\mathcal{F} = f \cdot g$ for some factor g.

Lemma 6. *Consider factor $\mathcal{G} = \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot \mathcal{H}$ where $\gamma_1,\ldots,\gamma_k$ are arbitrary factors, $f_X \in \mathcal{H}$ and $\mathtt{vars}(f_X) \subseteq \mathbf{S}_1,\ldots,\mathtt{vars}(f_X) \subseteq \mathbf{S}_k$ for some mechanism f_X of variable X. If $f_X \in \mathcal{F}$, then $\mathcal{F}\cdot\mathcal{G} = \mathcal{F}\cdot\mathcal{G}'$ where $\mathcal{G}' = \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot \sum_X \mathcal{H}$.*

Proof. Suppose $f_X \in \mathcal{F}$. Then $\mathcal{F} = f_X \cdot \mathcal{F}'$ for some factor $\mathcal{F}'$. Moreover, $\mathcal{F}\cdot\mathcal{G}$ equals to

$$
\begin{aligned}
&\mathcal{F}' \cdot f_X \cdot \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot \mathcal{H} \\
&= \mathcal{F}' \cdot \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot f_X \cdot \mathcal{H} \\
&= \mathcal{F}' \cdot \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot f_X \cdot \sum_X \mathcal{H} \quad \text{(by Theorem 3)} \\
&= \mathcal{F}' \cdot f_X \cdot \overset{=}{\sum}_{\mathbf{S}_1}\gamma_1 \cdots \overset{=}{\sum}_{\mathbf{S}_k}\gamma_k \cdot \sum_X \mathcal{H} \\
&= \mathcal{F}\cdot\mathcal{G}'. \quad \square
\end{aligned}
$$

We are now ready for the soundness proof. For jointree edge (i,j), let $\mathcal{M}'_{ij}$ and $\mathcal{M}'_{ji}$ denote the messages between i and j under separators $\mathbf{S}'$. We will next show $Pr(\mathbf{S}'_{ij}) = \mathcal{M}'_{ij}\mathcal{M}'_{ji}$ for all edges (i,j) by induction on rule applications. For each rule application, we will use $\mathbf{S}$ to denote the separators before thinning by the rule and $\mathbf{S}'$ to denote the separators after thinning by the rule. Initially, $Pr(\mathbf{S}_{ij}) = \mathcal{M}_{ij}\mathcal{M}_{ji}$ for all edges (i,j) by Theorem 5. We next show that this equality holds after each rule application. We consider three cases, one for each rule type.

(1) Rule (a) is applied to edge (i,j): $X \in \mathbf{S}_{ij}$ and $X \notin \mathbf{S}'_{ij}$. By definition of Rule (a) and Lemma 5, node i is strongly $(X,\mathbf{S})$-connected to some mechanism f_X hosted at leaf node l on the i-side and node j is strongly $(X,\mathbf{S})$-connected to some mechanism f_X hosted at leaf node r on the j-side.

First, we have $\mathcal{M}'_{ij}\cdot\mathcal{M}'_{ji} = \mathcal{M}_{ij}\cdot\mathcal{M}_{ji}$ by Corollary 1. Consider now any edge (k,z) on the path $l\ldots k\text{–}z\ldots r$ between leaf nodes l and r and suppose WLG that edge (i,j) is on the subpath $z\ldots r$. Using Lemma 6 with $\mathcal{F} = \mathcal{M}_{kz}$, $\mathcal{G} = \mathcal{M}_{zk}$ and $\mathcal{H} = \mathcal{M}_{ji}$, we get $\mathcal{M}_{kz}\cdot\mathcal{M}_{zk} = \mathcal{M}'_{kz}\cdot\mathcal{M}'_{zk}$. That is, removing X from the separator of edge (i,j) does not affect the product of messages for edge (k,z). Finally, consider any edge (k,z) that is not on the path between leaf nodes l and r. Let t be the node on this path which is closest to edge (k,z). Let l', r', u be the neighbors of t that are closest to l, r and edge (k,z), respectively. To show $\mathcal{M}_{kz}\cdot\mathcal{M}_{zk} = \mathcal{M}'_{kz}\cdot\mathcal{M}'_{zk}$, it suffices to show $\mathcal{M}'_{tu} = \mathcal{M}_{tu}$. WLG, suppose node j is closer to t than node i. Since $\mathcal{M}_{tu} = \overset{=}{\sum}_{\mathbf{S}_{tu}}\gamma\cdot\mathcal{M}_{l't}\cdot\mathcal{M}_{r't}$, where γ denotes the product of other invariant messages, we can use Lemma 6 again with $\mathcal{F} = \mathcal{M}_{l't}$, $\mathcal{G} = \mathcal{M}_{r't}$ and $\mathcal{H} = \mathcal{M}_{ij}$ to get $\mathcal{M}'_{l't}\cdot\mathcal{M}'_{r't} = \mathcal{M}_{l't}\cdot\mathcal{M}_{r't}$. Hence, applying Rule (a) preserves the product of messages for all jointree edges.

(2) Rule (b) is applied to edge (i,j): $X \in \mathbf{S}_{ij}$ and $X \notin \mathbf{S}'_{ij}$. By definition of Rule (b), node i is not a leaf and $X \notin \mathbf{S}_{ki}$ for $k \neq j$. Then $\mathcal{M}'_{ij} = \sum_X \mathcal{M}_{ij} = \mathcal{M}_{ij}$ since messages $\mathcal{M}_{ki}$ do not contain X. Moreover, $Pr(\mathbf{S}'_{ij}) = \sum_X Pr(\mathbf{S}_{ij}) = \sum_X \mathcal{M}_{ij}\cdot\mathcal{M}_{ji} = \mathcal{M}_{ij}\cdot(\sum_X \mathcal{M}_{ji}) = \mathcal{M}'_{ij}\cdot\mathcal{M}'_{ji}$. We next consider edges other than (i,j).

Since $\mathcal{M}'_{ij} = \mathcal{M}_{ij}$, all messages outgoing from node j are invariant. Hence, the product of messages is invariant for any edge on the j-side of edge (i,j). We next show that all message outgoing from i to neighbors $k \neq j$ are also invariant. This shows that the product of messages is also invariant for all edges on the k-side of any edge (k,i).

$$
\begin{aligned}
\mathcal{M}_{ik} &= \overset{=}{\sum}_{\mathbf{S}_{ik}}\mathcal{M}_{ji}\cdot\prod_{t\neq k, t\neq j}\mathcal{M}_{ti} \\
&= \overset{=}{\sum}_{\mathbf{S}_{ik}}(\sum_X \mathcal{M}_{ji})\cdot\prod_{t\neq k, t\neq j}\mathcal{M}_{ti} \\
&= \overset{=}{\sum}_{\mathbf{S}_{ik}}\mathcal{M}'_{ji}\cdot\prod_{t\neq k, t\neq j}\mathcal{M}'_{ti} \\
&= \mathcal{M}'_{ik}.
\end{aligned}
$$

The second step follows since $X \notin \mathbf{S}_{ki}$ and also $X \notin \mathbf{S}_{ti}$. The third step follows since messages $\mathcal{M}_{ti}$ are invariant to thinning variable X from separator $\mathbf{S}_{ij}$.

(3) Rule (c) is applied to edge (i,j). Similar to case (2).

1.9 PROOF OF THEOREM 13

Proof. Suppose replication $\mathcal{F}_1$ has width w. Then it must have a causal jointree $\mathcal{T}_1$ of width w. We can turn $\mathcal{T}_1$ into a causal jointree $\mathcal{T}_2$ for replication $\mathcal{F}_2$ by assigning more factors to leaf nodes in $\mathcal{T}_1$. In particular, for each replica $f_X \in \mathcal{F}_2 \setminus \mathcal{F}_1$, assign this replica to a leaf node in $\mathcal{T}_1$ which hosts a mechanism f_X. This guarantees that $\mathcal{T}_2$ will also have width w. Since replication $\mathcal{F}_2$ has a causal jointree of width w, its width must be $\leq w$. $\square$

1.10 PROOF OF THEOREM 14

Proof. The fact that Algorithm 1 computes a complete replication follows directly from the statement of the algorithm. Suppose there exists another complete replication $\mathcal{F}'$ that is different from the complete replication $\mathcal{F}$ computed by Algorithm 1. Then $\mathcal{F}'$ and $\mathcal{F}$ must differ on the number of mechanisms f_X for some functional variable X. Suppose X is the first variable visited by Algorithm 1 on which this disagreement takes place. Then $\mathcal{F}$ and $\mathcal{F}'$ must have the same number of X-feedings factors; otherwise, they will have a different number of mechanisms for some descendant of variable X. Since $\mathcal{F}$ and $\mathcal{F}'$ both satisfy Definition 13 and they have the same number of X-feeding factors, they must have the same number of mechanisms f_X. This contradicts the assumption that $\mathcal{F}'$ and $\mathcal{F}$ differ on the number of mechanisms for variable X. Hence, $\mathcal{F} = \mathcal{F}'$. $\square$

1.11 PROOF OF THEOREM 15

Lemma 7. *Consider a replication $\mathcal{F}$ that contains $n > 1$ mechanisms for variable X and m X-feeding factors where $n > m$. Let $\mathcal{F}'$ be the result of removing one mechanism for X from $\mathcal{F}$ (hence, $|\mathcal{F}'| = |\mathcal{F}| - 1$). The width of $\mathcal{F}'$ is no greater than the width of $\mathcal{F}$.*

Proof. Let $\mathcal{T}$ be a causal jointree for $\mathcal{F}$. We can turn $\mathcal{T}$ into a causal jointree $\mathcal{T}'$ for $\mathcal{F}'$ with no greater width as follows. Suppose there exists a leaf node in $\mathcal{T}$ that hosts two mechanisms f_X. We can then remove one of these mechanisms from the leaf without increasing the width. Suppose now that each leaf node in $\mathcal{T}$ hosts at most one mechanism f_X. The edges of $\mathcal{T}$ which contain variable X in their (thinned) separators form a set of connected subtrees. Each one of these subtrees will contain at most one mechanism f_X (otherwise thinning Rule (a) will apply). By the pigeonhole principle, at least one of these subtrees must contain a mechanism f_X but no X-feeding factors. We can show that removing this mechanism f_X from the subtree would not increase the causal width of resulting tree. Hence, if $\mathcal{F}$ has a causal jointree of width w, then $\mathcal{F}'$ has a causal jointree of no greater width. This implies that the width of $\mathcal{F}'$ is no greater than the width of $\mathcal{F}$. $\square$

Proof of Theorem 15. We first construct a new replication $\mathcal{G} = \mathcal{F} \cup \mathcal{F}'$. By Theorem 13, the width of $\mathcal{G}$ is no greater than the width of $\mathcal{F}'$. We then go through the functional variables in reverse-topological order (as visited by Algorithm 1). For each variable X, we compare if the X-mechanisms in $\mathcal{F}$ and $\mathcal{G}$ are equal. If so, we proceed to the next functional variable in the order. Otherwise, $\mathcal{G}$ contains more mechanisms than $\mathcal{F}$ and thus there are more X-mechanisms than X-feeding factors in $\mathcal{G}$. By Lemma 7, we are licensed to remove the excess X-mechanisms from $\mathcal{G}$ without increasing its width. By the end of this process, $\mathcal{G}$ will become equal to complete replication $\mathcal{F}$. Hence, the width of $\mathcal{F}$ is no greater than the width of $\mathcal{F}'$. $\square$

1.12 PROOF OF THEOREM 16

Proof. In a complete replication, the number of mechanisms for a functional variable is upper-bounded by the total number of mechanisms of its children. Let X be a functional variable and $f(X)$ be the number of mechanisms in the complete replication, then $f(X) = \sum_{C_i} f(C_i) \leq c \cdot max_{C_i} f(C_i)$ where C_i are the children of X in the DAG. Hence, we can recursively bound the number of mechanisms for all the functional nodes in a functional chain. Since the longest functional chain has a length of k, we can recursively apply the above bound for at most k steps. When the recursion terminates, $max_{C_i} f(C_i) = 1$ since all $f(C_i)$ will be non-functional variables. Therefore, $f(X) \leq c^k$ for each functional variable X in the replication. The inequality also holds for non-functional variables, $f(X) = 1 = c^0 \leq c^k$. Given a total of n nodes in the DAG, we have at most $O(nc^k)$ factors in the complete replication. $\square$

2 FURTHER EXPERIMENTS

We report here an additional experiment that reveals the importance of replication strategies and how such strategies interact with jointree construction methods.

We first note that classical methods for constructing jointrees do not directly apply to the construction of jointrees for replications. To see why, consider a set of factors $\mathcal{F}$ with no replicas. The classical method for constructing a jointree for such factors is to first construct a *primal graph.* This is a graph with nodes corresponding to the variables in factors $\mathcal{F}$ and which includes an edge between two variables iff they appear in the same factor.[1] Consider now a replication $\mathcal{F}'$ of factors $\mathcal{F}$. It follows immediately that the primal graph of $\mathcal{F}'$ is precisely the primal graph of $\mathcal{F}$. Hence, a classical jointree construction method will produce the

[1]There are various methods for constructing a jointree based on a primal graph; see [Darwiche, 2009, Ch 9]. One of the popular methods is to construct a low-width elimination order for the primal graph using the *minfill* heuristic and to then convert the order into a jointree of no greater width. This is what we used.

% func	replication type							
	c	$5c$	$10c$	$15c$	$20c$	$25c$	$30c$	complete
25%	126.2	334.1	594.6	854.3	1114.1	1373.6	1634.2	181.1
50%	151.2	561.4	1076.0	1588.9	2104.1	2618.3	3132.3	513.4
75%	178.7	799.3	1575.6	2352.9	3128.2	3907.0	4682.3	2564.3
100%	187.0	867.7	1721.0	2573.5	3429.6	4280.1	5132.8	4448.0

Table 1: Average size of replications. Replication nc means: the number of replicas for a node X is between $(n-1)c$ and nc where c is the number of children for node X.

Figure 1: Illustrating the impact of replication strategies.

same jointree for factors $\mathcal{F}$ and for all their replications. [Darwiche, 2020] proposed a jointree construction method that targets complete replications. For a non-leaf functional variable X, the method uses a distinct name for X in each of its n replicas and these distinct names are also used in the X-feeding factors whose count is also n. A jointree is then constructed using a classical technique followed by a reversal of the renaming process. While this method proved generally effective, it applies only to complete replications.

The experiment we conducted compared the complete replication strategy with random replications of increasing size, while varying the percentage of functional, non-root nodes in a Bayesian network (25, 50, 75, 100). The comparison was based on constructing jointrees using the minfill heuristic (see Footnote 1). We used the method of [Darwiche, 2020] for complete replications, and adapted it somewhat arbitrarily for random replications. In particular, when the number of X-feeding factors did not match the number of X-mechanisms, we renamed variables in the X-feeding factors distinctly to the extent possible and randomly thereafter.

Table 1 shows the size of random and complete replications, with some random replications being larger than complete replications. Figure 1 shows the mean maximal cluster size (width+1) for jointrees and causal jointrees where each data point is an average over 100 random Bayesian networks, each containing 100 nodes. A few patterns are clear. First, the causal width is always smaller than the width, and quite substantially smaller, even when using random replications. Second, complete replications always produced a smaller causal width compared to random replications, particularly when the number of functional nodes is largest (100%). Third, increasing the size of a random replication almost always correlated with decreasing the causal width but up to a certain point after which increasing the size of a replication did not help. The few exceptions to this pattern highlight the suboptimality of the jointree construction method we used (see Theorem 13) and the suboptimality of the heuristic for applying thinning rules. Beyond emphasizing some of the theoretical results we presented earlier, this experiment further highlights the practical significance of causal treewidth and causal jointrees as they can lead to an exponential reduction in inference complexity. The experiment also highlights the need for developing principled jointree construction methods that target replications which are not complete, and highlight the need for further heuristics to guide the application of thinning rules.