Temporal Properties of Conditional Independence in Dynamic Bayesian Networks

Anonymous submission

Abstract

Dynamic Bayesian networks (DBNs) are compact graphical representations used to model probabilistic systems where interdependent random variables and their distributions evolve over time. In this paper, we study the verification of the evolution of conditional-independence (CI) propositions against temporal logic specifications. To this end, we consider two specification formalisms over CI propositions: linear temporal logic (LTL), and non-deterministic Büchi automata (NBAs). This problem has two variants. Stochastic CI properties take the given concrete probability distributions into account, while structural CI properties are viewed purely in terms of the graphical structure of the DBN. We show that deciding if a stochastic CI proposition eventually holds is at least as hard as the Skolem problem for linear recurrence sequences, a long-standing open problem in number theory. On the other hand, we show that verifying the evolution of structural CI propositions against LTL and NBA specifications is in PSPACE, and is NP- and coNP-hard. We also identify natural restrictions on the graphical structure of DBNs that make the verification of structural CI properties tractable.

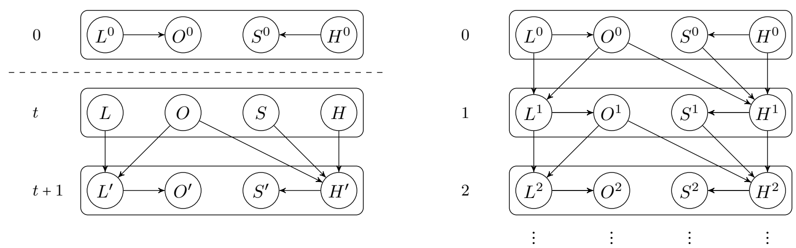
1 Introduction

Bayesian networks (BNs) (Pearl 1985, 1988a; Neapolitan 1989) are prominent tools in both data science and artificial intelligence that enable modeling and reasoning under uncertainty. BNs succinctly represent a full joint probability distribution by using a directed acyclic graph (DAG) as a template to capture dependencies between variables and prescribe the probability distribution of each variable conditioned on its parents. BNs have successfully been applied in medical AI (Lucas, van der Gaag, and Abu-Hanna 2004)), computer vision (Wu et al. 2007), natural language processing (Manning and Schütze 1999), robotics (Thrun, Burgard, and Fox 2005), fault diagnosis (Jemal et al. 2003), bioinformatics (Friedman 2000), and risk assessment (Fenton and Neil 2012).

Dynamic Bayesian Networks (DBNs) extend BNs to describe systems where the outcomes modeled by random variables evolve with time (Murphy 2002; Koller and Friedman 2009). DBNs succinctly represent a *sequence* of full joint probability distributions of a set of random variables, i.e., a DBN prescribes an initial joint probability distribution for the variables $\mathbf{V}^0$, and also prescribes the joint distribution of $\mathbf{V}^{t+1}$, the variables at time step (or time slice) $t+1$, conditioned on the variables $\mathbf{V}^t$ at time step t. These are respectively given by an *initial* BN and a *step* BN, and their corresponding DAGs are collectively referred to as the *DBN template*. To make a concrete DBN, the template is instantiated with *conditional probability distributions* (CPDs). The temporal dimension of DBNs has motivated applications in robotics (Thrun, Burgard, and Fox 2005), bioinformatics (Friedman 2000), systems biology (Palaniappan and Thiagarajan 2012), engineering and fault diagnosis in dynamical systems (Meng et al. 2024; Jemal et al. 2003), and speech recognition (Zweig and Russell 1998).

Example 1.1. To illustrate DBNs and DBN-templates, consider a system coordinating different probabilistic components either having access only to *low-security* information or also to *high-security* information. In each time step t, the low-security components provide an input L^t and the high-security components provide an input H^t. The system then produces a low-security output O^t and a high-security output S^t (S for secret). The dependencies between these variables are depicted in the DBN-template depicted in Figure 1a: The *initial template* marked with 0 expresses that initially O^0 depends only on L^0 and S^0 depends only on H^0. All other pairs of variables are independent. The *step template* depicted below uses the variables L, O, S, and H representing the variables at the current time step as well as copies L', O', S', and H' representing the variables at the next time step. For example, this template captures that the next low-security input L^{t+1} always depends directly only on the previous low-security input L^t and output O^t. To represent all (direct) dependencies between variables in the timed sequence of variables, the template can be unfolded into one infinite directed acyclic graph (DAG), called the *unfolding of the DBN-template*, as depicted in Figure 1b.

A DBN based on this DBN-template additionally consists of the CPDs for each variable in the initial template and for each primed variable in the step template given its parents. Assuming that all variables take only values 0 and 1, an example CPD for the variable L' in the step template is depicted in Figure 1c. For each combination of values of the parent variables L and O, it specifies the probability with which L' takes value 0 and 1. This CPD is applied at each time step in the unfolding of the DBN. For variables without parents, such as L^0 and H^0 in the initial template, the CPDs



(a) Example of a DBN-template. (b) Unfolding of the DBN-template.

		L'	
L	O	0	1
0	0	0.2	0.8
0	1	0.5	0.5
1	0	0.3	0.7
1	1	0	1

(c) Example of a CPD.

Figure 1: The DBN-template described in Example 1.1 and its unfolding as well as an example CPD.

specify the probabilities with which they take value 0 and 1.

A fundamental concern (see, e.g., (Heckerman, Geiger, and Chickering 1995)) in reasoning about probabilistic models is the characterization and/or deduction of stochastic conditional independence (CI) of sets $\mathbf{X}$ and $\mathbf{Y}$ of random variables given the values of a set $\mathbf{Z}$, denoted $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$.

In seminal work, (Geiger, Verma, and Pearl 1990) show that the truth of these CI propositions, which satisfy the graphoid axioms, can be deduced from the underlying DAG template in the case of BNs. Specifically, they define structural conditional independence through the efficiently testable graphical notion of *d-separation*, denoted as $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ when $\mathbf{Z}$ d-separates $\mathbf{X}$ and $\mathbf{Y}$. They then show soundness: if $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$, then $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$ for all conditional probability distributions. Subsequently, Meek (1995) showed a form of completeness: i.e., if $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ fails then $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$ also fails for all but a (Lebesgue) measure-0 set of conditional probability distribution parameters.

DBNs can naturally be associated with an infinite sequence of BNs, with the t-th term being obtained by unfolding up to time slice t. We call a statement of the form $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ (respectively, $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$) an atomic proposition of structural (respectively, stochastic) CI, and say that it holds at time t if $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ (respectively, $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$) holds in the unfolding of the DBN up to time t. Given a collection A of structural (respectively, stochastic) CI propositions, the DBN defines a *trace*, i.e., an infinite word over the alphabet 2^A whose t-position records which of the propositions hold at time t.

In this paper, we concern ourselves with checking the properties of the trace, such as: is $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ ever false? In Example 1.1, is the output always independent of the secret given the low-security input? In a system, is it always the case that if inputs I_1, I_2 are independent, then so are outputs O_1, O_2?

To express temporal properties of systems, the use of temporal logics such as linear temporal logic (LTL) and of non-deterministic Büchi automata (NBAs), capturing all ω-regular languages, has emerged as a success story over the past decades (see, e.g., (Baier and Katoen 2008)). We aim to employ these formalisms to talk about the temporal aspects of CIs.

Example 1.2. The three properties mentioned above are expressed in LTL as: (i) $\Diamond \neg(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$, where $\Diamond$ is the temporal modality for "eventually"; (ii) $\Box(O \perp\!\!\!\perp S \mid L)$, where $\Box$ is the temporal modality for "globally", and dual to $\Diamond$; (iii) $\Box((I_1 \perp\!\!\!\perp I_2) \to (O_1 \perp\!\!\!\perp O_2))$.

Given an LTL formula φ over the set of atomic propositions A or an NBA $\mathcal{B}$ over the alphabet 2^A, the *structural CI model-checking problem for DBN-templates* now asks whether the trace of a DBN-template satisfies φ or is accepted by $\mathcal{B}$, respectively. The *stochastic CI model-checking problem for DBNs* asks the analogous question for the trace of a DBN with respect to a set of stochastic CI propositions.

1.1 Contributions

1. In Section 3, we introduce temporal specification mechanisms for the evolution of structural or stochastic CI propositions in DBN-templates and DBNs, respectively, using LTL and NBAs. We formulate the resulting structural and stochastic CI model-checking problems.
2. In Section 4, we show that the structural CI model-checking problems of DBN-templates against LTL formulas and against NBAs are both in PSPACE and NP-hard as well as coNP-hard. Under the natural restriction that the initial template of a DBN-template only contains edges that also appear as intra-slice edges in the step template, we prove that the problems are in P.
3. Given full DBNs with CPDs, we show in Section 5 that checking eventual stochastic CI is as hard as the Skolem problem for linear recurrence sequences, a famous number-theoretic problem whose decidability status has been open for many decades. This implies that a decidability result for the stochastic CI model-checking problems is out of reach without a breakthrough in analytic number theory.

1.2 Related work

The question how to detect structural CIs in BNs has been answered in the 1980s and 1990s by showing that d-separation characterizes all structural CIs that follow from the structure of a BN, that this is equivalent to stochastic CI under all choices of CPDs, and by showing that the d-separation can compute these structural CIs in polynomial time (see (Pearl 1988b; Geiger, Verma, and Pearl 1990; Meek 1995)). Exactly determining whether a stochastic CI holds requires exact computation of the necessary conditional probability distributions. Methods for approximate testing of conditional independence of discrete random variables, however, are an active area of research (see, e.g., (Canonne et al. 2018; Teymur and Filippi 2020)). Orthogonally, seminal work by Boutilier et al. (1996) studies so-called context-sensitive independence expressing that variables might only be independent under specific assignments of values to other variables. Like stochastic CI, this kind of independence depends on the concrete CPDs.

We are not aware of thorough studies of d-separation and the detection of CIs in DBNs, let alone the formal verification of temporal properties of CIs in DBNs. Regarding other extension of BNs, Shen et al. (2019) study CIs in testing BNs, an extension of BNs representing a set of probability distributions instead of a singly distributions, and show that d-separation can still be used to detect structural CIs.

2 Preliminaries

Probability spaces and conditional independence. We assume knowledge of the basics of probability theory (Klenke 2007), and record the relevant prerequisites in the technical appendix for completeness. In this paper, we work with discrete random variables. Disjoint tuples of random variables $\mathbf{X}, \mathbf{Y}$ are considered conditionally independent given $\mathbf{Z}$ (denoted as $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$), if for any values $\mathbf{x}, \mathbf{y}, \mathbf{z}$ (provided $\Pr[\mathbf{Z} = \mathbf{z})] > 0$, the following holds: $\Pr(\mathbf{X} = \mathbf{x}, \mathbf{Y} = \mathbf{y} \mid \mathbf{Z} = \mathbf{z}) = \Pr(\mathbf{X} = \mathbf{x} \mid \mathbf{Z} = \mathbf{z}) \cdot \Pr(\mathbf{Y} = \mathbf{y} \mid \mathbf{Z} = \mathbf{z})$.

Bayesian networks A Bayesian network (BN) is a type of probabilistic graphical model that expresses a set of variables and their conditional dependencies using a directed acyclic graph (DAG), where each node represents a variable, and the edges indicate direct probabilistic dependencies between the variables.

Definition 2.1 (Bayesian Network)**.** A Bayesian network (BN) over a finite set $\mathbf{V}$ of discrete random variables is a tuple $B = \langle \mathbf{V}, \mathcal{E}, \mathcal{P} \rangle$, where:

- Each element of $\mathbf{V}$ is represented as a vertex of a DAG;
- The set of directed edges is $\mathcal{E} \subseteq \mathbf{V} \times \mathbf{V}$; we call the DAG $\langle \mathbf{V}, \mathcal{E} \rangle$ the *template* of the BN.
- The probability distribution of $\mathbf{V}$ is expressed in terms of a collection $\mathcal{P}$ of conditional probability distributions (CPDs), i.e., for each variable X with parents $\mathsf{pa}(X) = \mathbf{Y}$, $\mathcal{P}$ prescribes $\Pr[X = x \mid \mathbf{Y} = \mathbf{y}]$ for all possible $x, \mathbf{y}$.

We refer to the set of all BNs with a given BN-template $\mathcal{T}$ as the family $\mathsf{Fam}(\mathcal{T})$.

As an illustrative scenario, consider a BN where all variables are binary. Then $\mathcal{P}$ consists of $\sum_{X \in \mathbf{V}} 2^{|\mathsf{pa}(X)|}$ parameters, each term of the summation counting the number of parameters required to prescribe the probability of X being 1, depending on the values taken by its parents.

Definition 2.2 (d-paths and d-separation)**.** Given a BN-template (i.e., DAG) $\mathcal{T} = (\mathbf{V}, \mathcal{E})$ and three pairwise disjoint sets $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \subseteq \mathbf{V}$ of nodes, a d-path from $\mathbf{X}$ to $\mathbf{Y}$ with respect to $\mathbf{Z}$ is a sequence $W_0, \ldots, W_k$ of nodes with the following properties:

- $W_0 \in \mathbf{X}$ and $W_k \in \mathbf{Y}$,
- for each $i < k$, either $(W_i, W_{i+1}) \in \mathcal{E}$ or $(W_{i+1}, W_i) \in \mathcal{E}$
- for all $0 < i < k$, if the node W_i has an outgoing edge to W_{i-1} or W_{i+1} (or both), then $W_i \notin \mathbf{Z}$,
- for all $0 < i < k$, if the node W_i has incoming edges from both W_{i-1} and W_{i+1}, then one of the descendants of W_i is in $\mathbf{Z}$ (we consider a node to be its own descendant and ancestor). We call such a node W_i a *collider* and say that the collision is *attributed* to the descendants of W_i in $\mathbf{Z}$.

If there is no such path, we say that $\mathbf{Z}$ d-separates $\mathbf{X}$ and $\mathbf{Y}$. In this case, we write $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ and say that $\mathbf{X}$ and $\mathbf{Y}$ are *structurally independent* given $\mathbf{Z}$.

We remark that both structural and stochastic conditional independence (CI) satisfy the *graphoid* axioms (Geiger, Verma, and Pearl 1990, p. 511, (4a)-(4d)) (see also (Spohn 1980) for a proof of the stochastic case). Structural CI via d-separation in a BN-template $\mathcal{T}$ is known to be equivalent to stochastic CI in all members of the family $\mathsf{Fam}(\mathcal{T})$.

Theorem 2.3 (Soundness and completeness of d-separation, (Pearl 1988b), (Meek 1995))**.** *Given a BN-template $\mathcal{T} = \langle \mathbf{V}, \mathcal{E} \rangle$ and pairwise disjoint sets of random variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \subseteq \mathbf{V}$, the following two statements are equivalent and can be checked in polynomial time:*

- $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$,
- *for all BNs in* $\mathsf{Fam}(\mathcal{T})$*, we have* $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$*, i.e.,* $\mathbf{X}$ *and* $\mathbf{Y}$ *are (stochastically) independent given* $\mathbf{Z}$*.*

Dynamic Bayesian networks (DBNs). We use Dynamic Bayesian networks (Murphy 2002; Koller and Friedman 2009) to model probabilistic systems where the entities captured by random variables evolve over time. Formally, consider a finite set $\mathbf{V}$ of random variables. We track the evolution of $\mathbf{V}$ with time through a countably infinite sequence $(\mathbf{V}^t)_{t=0}^{\infty}$ of copies of the random variables in $\mathbf{V}$.

The evolution itself is modeled as a dynamical system whose state at time t is a BN involving the variables $\bigcup_{i=0}^{t} \mathbf{V}^t$. The initial BN is given by $\langle \mathbf{V}^0, \mathcal{E}^0, \mathcal{P}^0 \rangle$.

We use a copy $\mathbf{V}'$ of the random variables to express the update dynamics, which at each step $t \geq 1$, introduce the variables $\mathbf{V}^t$ with parents in $\mathbf{V}^{t-1} \cup \mathbf{V}^t$, while keeping the rest of the network unchanged. Formally, we have a *step template* which is a BN-template with variables $\mathbf{V} \cup \mathbf{V}'$ and edge relation $\mathcal{E}^{\mathsf{step}} \subseteq (\mathbf{V} \cup \mathbf{V}') \times \mathbf{V}'$. Semantically, if X' has parents $Y'_{i_1}, \ldots, Y'_{i_\ell}, Y_{j_1}, \ldots, Y_{j_k}$ in the step template, then for each $t \geq 1$, X^t has parents $Y^t_{i_1}, \ldots, Y^t_{i_\ell}, Y^{t-1}_{j_1}, \ldots, Y^{t-1}_{j_k}$. Finally, we have the conditional probability distribution parameters $\mathcal{P}^{\mathsf{step}}$ which prescribe $\Pr[X' = x \mid \mathsf{pa}(X') = \mathbf{y}]$ for all $x, \mathbf{y}$. Semantically, we have that the distribution of X^t conditioned on its parents is the same for all $t \geq 1$.

We can thus specify a DBN via $\langle \mathbf{V}, \mathcal{E}^0, \mathcal{P}^0, \mathcal{E}^{\mathsf{step}}, \mathcal{P}^{\mathsf{step}} \rangle$. Its structural properties are given by the tuple $\mathcal{T}_{\mathsf{DBN}} = \langle \mathbf{V}, \mathcal{E}^0, \mathcal{E}^{\mathsf{step}} \rangle$, which we refer to as a *DBN-template*. Analogous to BNs, we refer to the set of all DBNs with a given template $\mathcal{T}_{\mathsf{DBN}}$ as the family $\mathsf{Fam}(\mathcal{T}_{\mathsf{DBN}})$. As explained earlier, in a DBN of binary variables, $\mathcal{P}^0$ consists of $\sum_{X \in \mathbf{V}^0} 2^{|\mathsf{pa}(X)|}$ parameters, and $\mathcal{P}^{\mathsf{step}}$ consists of $\sum_{X \in \mathbf{V}'} 2^{|\mathsf{pa}(X)|}$ parameters.

We say that a DBN-template is *restricted* if whenever $(X^0, Y^0) \in \mathcal{E}^0$, we also have that $(X', Y') \in \mathcal{E}^{\mathsf{step}}$. So, the dependencies that exist initially at time step 0 also have to be present at all later time steps, which is captured by their presence in the step template between the corresponding primed variables. The DBN-template depicted in Figure 1 is an example of a restricted DBN template.

Equivalence of DBNs with Markov Chains. That the semantics of a DBN can be expressed in terms of the evolution of a Markov chain is folklore. For completeness, we state two elementary lemmas expressing this fact formally and defer the constructions that establish them to the appendix.

Lemma 2.4. *Given a DBN with k binary random variables, we can construct an equivalent Markov chain with 2^k states.*

Given a Markov chain with K states, we can construct an equivalent DBN with $\lceil \log K \rceil$ binary random variables.

3 Specification formalisms for temporal conditional independence properties

In this paper, we study the verification of DBNs and DBN-templates against linear-temporal properties regarding the evolution of conditional independencies (CIs) over time. For DBN-templates with variables $\mathbf{V}$ we use atomic propositions of the form $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$, which we call *structural CI propositions*. We say that $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ holds in DBN-template $\mathcal{T}$ at time t if $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ holds in the unfolding of $\mathcal{T}$ to a BN-template after t time steps. Similarly, for full DBNs with concrete CPDs, we use *stochastic CI propositions* $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$ that hold at time t in a DBN $\mathcal{B}$ if $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ holds in the BN formed at time t. A first result connecting structural CI and stochastic CI is immediate (for completeness, the proof can be found in the appendix):

Proposition 3.1. *If $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ in a DBN-template $\mathcal{T}$ for some t, then, $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ in every DBN $\mathcal{B} \in \mathrm{Fam}(\mathcal{T})$.*

The converse direction, i.e., the completeness of d-separation for DBNs, however, turns out to be intricate. We discuss this issue in Section 6.

Remark 3.2. The *finite-horizon* problem, that is, deciding whether a structural or stochastic CI statement holds at a fixed time t, can be solved in time polynomial in t and in the size of the DBN(-template) via unfolding. Although solving it in logarithmic time remains an open question, this paper focuses on *infinite-horizon* problems concerning the full temporal evolution.

Trace of a DBN(-template). For any (finite) set A of such structural CI propositions, a DBN-template defines a *trace* $\tau \in (2^A)^\omega$, i.e., an infinite word over the alphabet 2^A, where the letter at position t indicates which propositions hold at time t. Likewise, a full DBN defines a trace $\tau \in (2^B)^\omega$ for any (finite) set B of stochastic CI propositions.

We seek to verify whether the trace τ of a DBN-template or of a DBN satisfies a logical specification. We use the notation $\tau(t)$, to refer to the t-th position of τ, and the notation $\tau[t:\infty]$ to refer to the suffix of τ starting at position t, e.g., $\tau[0:\infty] = \tau$, $\tau[t:\infty][t':\infty] = \tau[t+t':\infty]$.

Linear temporal logic (LTL). We consider two common logical formalisms (for a comprehensive exposition of which the reader is referred to (Baier and Katoen 2008)). The first is linear temporal logic (LTL, introduced in (Pnueli 1977)), whose formulae φ over a set of atomic propositions A are syntactically given by the grammar $\varphi := a \mid \neg\varphi \mid \varphi \wedge \varphi \mid \bigcirc \varphi \mid \varphi \,\mathcal{U}\, \varphi$ where $a \in A$ is an atomic proposition. The operator $\bigcirc$ is called "next" and the operator $\mathcal{U}$ is called "until". Semantically, it is defined recursively whether an infinite word τ over 2^A satisfies an LTL formula (written $\tau \models \varphi$) as follows:

- $\tau \models a$ if and only if $a \in \tau(0)$,
- $\tau \models \neg\varphi$ if and only if $\tau \not\models \varphi$,
- $\tau \models \varphi_1 \wedge \varphi_2$ if and only if both $\tau \models \varphi_1$ and $\tau \models \varphi_2$,
- $\tau \models \bigcirc\varphi$ if and only if $\tau[1:\infty] \models \varphi$,
- $\tau \models \varphi_1 \,\mathcal{U}\, \varphi_2$ if and only if there exists t s.t. $\tau[t:\infty] \models \varphi_2$ and for all $t' < t$, $\tau[t':\infty] \models \varphi_1$.

For notational convenience, we allow access to all the usual Boolean connectives, true, false, as well as the temporal modalities $\Diamond$ ("eventually"; $\Diamond\varphi$ is equivalent to true $\mathcal{U}\,\varphi$) and its dual $\Box$ ("globally"; $\Box\varphi$ is equivalent to $\neg\Diamond\neg\varphi$).

E.g., consider the DBN-template given in Example 1.1. We can use d-separation to argue that the structural formula $\Box(O \perp\!\!\!\perp S \mid L)$ holds, i.e., $(O^t \perp\!\!\!\perp S^t \mid L^t)$ holds for all t. Using the temporal LTL-operators, also more involved properties can be expressed: the formula $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y})\,\mathcal{U}\,\neg(\mathbf{Y} \perp\!\!\!\perp \mathbf{Z})$, e.g., expresses that the sets of variables $\mathbf{X}$ and $\mathbf{Y}$ are structurally independent at least until $\mathbf{Y}$ and $\mathbf{Z}$ are dependent.

For LTL, we investigate the following two problems:

- **Structural LTL model-checking of DBN-templates:** For a DBN-template $\mathcal{T}$ with variables $\mathbf{V}$ and an LTL formula φ over the set A of structural CI propositions using $\mathbf{V}$, decide whether the trace $\tau \in (2^A)^\omega$ of $\mathcal{T}$ satisfies φ.
- **Stochastic LTL model-checking of DBNs:** For a DBN $\mathcal{B}$ with variables $\mathbf{V}$ and an LTL formula φ over the set B of stochastic CI propositions using $\mathbf{V}$, decide whether the trace $\tau \in (2^B)^\omega$ of $\mathcal{B}$ satisfies φ.

We indeed employ d-separation as a tool to reason more generally about specifications involving only structural independence propositions in Section 4. On the other hand, we prove that evaluating stochastic formulae as simple as $\Diamond(X \perp Y)$ can be number-theoretically hard (Lemma 5.1) showing that a decidability result for stochastic LTL or NBA model-checking of DBNs is out of reach without a breakthrough in number theory.

Non-deterministic Büchi automata (NBAs). The second formalism we consider is that of nondeterministic Büchi automata (NBAs, introduced in (Richard Büchi 1966)), which express precisely the class of ω-regular temporal properties. An NBA is a tuple $\mathcal{A} = (Q, \Sigma, \Delta, Q_0, F)$ where Q is a finite set of states, Σ is the alphabet, $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation, $Q_0 \subseteq Q$ is the set of initial states and $F \subseteq Q$ is the set of accepting states. A run of $\mathcal{A}$ on an infinite word $\tau = w_0 w_1 w_2 \cdots \in \Sigma^\omega$ is a sequence $\rho = q_0 q_1 \ldots$ of states such that $q_0 \in Q_0$ and $(q_i, w_i, q_{i+1}) \in \Delta$ for each $i \in \mathbb{N}$. The run ρ is accepting if $q_j \in F$ for infinitely many $j \in \mathbb{N}$. We say that $\mathcal{A}$ accepts τ if there exists an accepting run on τ.

We remark that it is known that NBAs are strictly more expressive than LTL formulae, however translating an LTL formula into an equivalent NBA can lead to an exponential increase in size. The resulting NBA model-checking problems we consider are:

- **Structural NBA model-checking of DBN-templates:** For a DBN-template $\mathcal{T}$ with variables $\mathbf{V}$ and an NBA $\mathcal{A}$ over the alphabet 2^A where A is the set of structural CI propositions using $\mathbf{V}$, decide whether the trace $\tau \in (2^A)^\omega$ of $\mathcal{T}$ is accepted by $\mathcal{A}$.
- **Stochastic NBA model-checking of DBNs:** For a DBN $\mathcal{B}$ with variables $\mathbf{V}$ and an NBA $\mathcal{A}$ over the alphabet 2^B where B is the set of stochastic CI propositions over $\mathbf{V}$, decide if the trace $\tau \in (2^B)^\omega$ of $\mathcal{B}$ is accepted by $\mathcal{A}$.

4 Structural conditional independence

In this section, we study the properties of the trace $\tau \in (2^A)^\omega$ of a DBN-template with variables $\mathbf{V}$ with respect to a set A of structural CI propositions and show how to check whether the trace satisfies a logical specification. The main result we will establish is the following:

Main result 4.1. *The structural LTL and NBA model-checking problems for DBN-templates are in PSPACE and NP-hard as well as coNP-hard.*

For restricted DBN-templates, the structural LTL and NBA model-checking problems are in PTIME.

To prove this result, we will show that the trace of a DBN-template with respect to structural CI propositions is ultimately periodic by virtue of being represented as the run of a deterministic transition system with $2^{O(|\mathbf{V}|^2)}$ states. We shall further demonstrate that in this transition system, a state can be represented in $O(|\mathbf{V}|^2)$ space, the successor can be computed in time polynomial in $|\mathbf{V}|$, and the labeling function (i.e., which propositions hold in a given state) can be computed in time polynomial in $|\mathbf{V}|$, $|A|$. We then rely on the well-known results that checking whether all traces of an N-state transition system are accepted by a size K NBA can be done in nondeterministic space $O(\text{polylog } (N \cdot K))$ and whether all traces satisfy a size-M LTL formula φ can be done in nondeterministic space $O(\text{polylog } N \cdot \text{poly } M)$ (see (Vardi and Wolper 1986) and, e.g., (Baier and Katoen 2008, Proof of Lemma 5.47) for a detailed presentation).[1]

The improved complexity for restricted DBN-templates follows from the insight that the trace of a restricted DBN-template on variables $\mathbf{V}$ is constant from time $|\mathbf{V}|^2$ onwards.

To convey a feeling for the hardness results, we first provide an example of a family of DBN-templates whose traces have periods exponential in the number of used variables:

Example 4.2. The k-th DBN-template of this family has variables $X_0, X_1, \ldots, X_k$ as well as $W_{1,0}, W_{1,1}, W_{2,0}, \ldots, W_{k,0}, W_{k,1}, \ldots, W_{k,p_k-1}$, where p_i refers to the i-th prime. The initial template has edges of the form $(X^0_{i-1}, W^0_{i,0})$ and $(X^0_i, W^0_{i,0})$ for all i, and the step template has edges $(X_0, X'_0), (X_k, X'_k)$, and edges from $W_{i,r}$ to $W'_{i,r+1(\text{mod } p_i)}$. By construction, we have $(X_0 \perp\!\!\!\perp X_k \mid \{W_{1,0}, \ldots W_{k,0}\})$ if and only if the timestep t is divisible by all of $2, 3, \ldots, p_k$ (see Figure 2). Intuitively, we go from X_0 to X_k via "islands" $X_1, \ldots, X_{k-1}$. The

[1]Provided one can compute the label and successor of a state in the transition system in polylog N space, which, in our case, we show we can as $N = 2^{O(|\mathbf{V}|^2)}$ and so polylog N = poly $|\mathbf{V}|$.

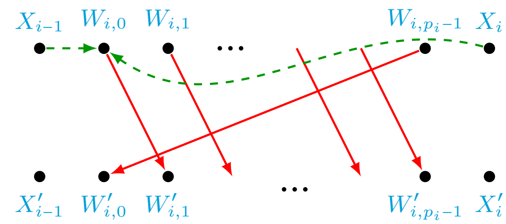


Figure 2: Bridge between islands X_{i-1} and X_i. Initial template in dashed green edges.

"bridge" between successive islands X_{i-1}, X_i uses the edges of the initial template, and is open precisely when t is divisible by p_i, i.e., a collision can be attributed to $W^t_{i,0}$. We need all bridges to be open simultaneously to make the journey: this gives us a period $2 \cdot 3 \cdot \cdots \cdot p_k$, which is exponential in $|\mathbf{V}| = (k+1) + 2 + 3 + \cdots + p_k$.

We establish NP-hardness of deciding whether $\Diamond(X \perp\!\!\!\perp Y \mid \mathbf{Z})$ holds: we reduce from the NP-complete intersection problem for unary DFA (Blondin and Mckenzie 2014). Since we also have access to the negated formula, we can also reduce from the complementary problem. Further, both properties can be expressed by fixed NBAs, and we get the following result whose proof is an adaptation of the example above, and given in the technical appendix.

Lemma 4.3. *The LTL and NBA model-checking problems with structural-independence propositions are hard for NP as well as for coNP.*

4.1 DBN traces through transition systems

We shall now prove the PSPACE upper bounds for the structural model-checking problems by showing that any trace of a DBN-template with variables $\mathbf{V}$ and structural independence propositions A can be obtained as the run of a deterministic transition system with $2^{O(|\mathbf{V}|^2)}$ states, each of whose states can be represented in $O(|\mathbf{V}|^2)$ space, whose successor function can be computed in time polynomial in $|\mathbf{V}|$, and whose labelling function can be computed in time polynomial in $|\mathbf{V}|, |A|$.

We start by making a key observation about d-paths:

Claim 1: If there exists a d-path from X to Y with respect to a set $\mathbf{Z}$ of size k in an arbitrary BN, there is one with at most k collisions, all of which occur inside $\mathbf{Z}$.

The above claim (see technical appendix for proof) motivates us to compute the pairs in $\{X^t, Y^t, Z^t_1, \ldots, Z^t_k\}$ that are connected by collision-free paths in order to determine whether there is a d-path relative to $\mathbf{Z}^t$ from X^t to Y^t. However, we must be slightly careful: a path that concatenates collision-free paths via edges (Z^t, X^t) and (Z^t, Y^t) is actually blocked by Z^t.

Construction of the transition system. We construct a transition system $\mathcal{S} = (Q, q_0, \to, A, L)$, where Q will be the state space with an initial state q_0, $\to$ a deterministic successor relation, A the set of structural CI propositions, and $L: Q \to 2^A$ the labeling function. The states in S will be BN-templates that represent the connections via collision-free d-

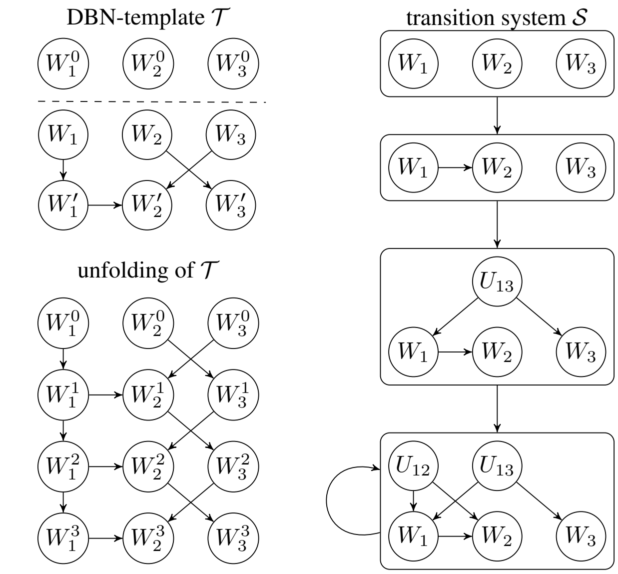


Figure 3: An example of a DBN-template $\mathcal{T}$ with its unfolding and the transition system $\mathcal{S}$ constructed from $\mathcal{T}$. Variables U_{ij} without outgoing edges are omitted. Note, e.g., that the variable U_{13} is connected to W_1 and W_2 after two time steps reflecting that there is a collision-free d-path $W_1^2, W_1^1, W_2^1, W_3^2$ connecting W_1^2 and W_3^2 in the unfolding through the previous time slices.

paths in the DBN-template at some time t. For an illustration of all steps of the construction, see Figure 3.

State space and labeling function: The representative BN-template for time $t = 0$ is simply the initial BN-template. The representative BN-templates for times $t > 0$ use variables $\mathbf{V} = \{W_1, \dots, W_n\}$, and auxiliary variables $\mathbf{U} = \{U_{ij} : 1 \le i, j \le n\}$, corresponding to unordered pairs of distinct i, j. The edge (W_i, W_j) is always present if and only if (W_i', W_j') is an edge in the step-template BN. In the representative at time t, we additionally draw edges (U_{ij}, W_i) and (U_{ij}, W_j) if there is a collision-free d-path from W_i^t to W_j^t in the original DBN-template, and the intermediate vertices along this path do not belong to $\mathbf{V}^t$. Note how the latter requirement rules out the possibility of a path being blocked by an observed variable. So, by definition, $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ holds at time t in the DBN if and only if $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ holds in the representative BN-template; running d-separation queries on the latter settles the issue of the labelling function, which hence can be computed in time polynomial in $|\mathbf{V}|$ and A.

We observe that for $t > 0$, there are $2^{\binom{|\mathbf{V}|}{2}}$ possible representatives, depending on the choice of which U_{ij} are made parents, each representable as a graph with $O(|\mathbf{V}|^2)$ vertices. This establishes the size requirements described at the beginning of Section 4.

Transition relation: Above, the transition relation of the reachable part of the state space is implicitly given by following the time steps. Now, we describe how to compute the successor of a representative BN-template $\mathcal{B}$ on $\mathbf{V} \cup \mathbf{U}$ directly in polynomial time. For this, consider the graph with vertices $\mathbf{U} \cup \mathbf{V} \cup \mathbf{V}' \cup \mathbf{U}'$. Draw edges in the subgraph induced by $\mathbf{U} \cup \mathbf{V}$ as prescribed by $\mathcal{B}$, and edges in $(\mathbf{V} \cup \mathbf{V}') \times (\mathbf{V}')$ as prescribed by the step template. Now, for each $i \neq j$, we add edges from U_{ij}' to W_i' and W_j' if there is an $A \in \mathbf{U} \cup \mathbf{V}$ that can reach W_i' and W_j' in the graph constructed so far without using edges inside $\mathbf{V}'$.

The correctness of this construction can be seen as follows: If A is in $\mathbf{V}$, this is a collision-free d-path from W_i' to W_j'. If $A \in \mathbf{U}$ it has edges to some $B_i \in \mathbf{V}$ reaching W_i' and $B_j \in \mathbf{V}$ reaching W_j'. The edges to B_i and B_j represent a collision-free d-path which can then be extended to a collision-free d-path from W_i' to W_j'. Finally, we restrict the graph to $\mathbf{V}' \cup \mathbf{U}'$ to obtain the successor of $\mathcal{B}$. Clearly, this successor can be computed in time polynomial in $|\mathbf{V}|$.

PSPACE upper bound: So, we have constructed the deterministic transition system satisfying the requirements described at the beginning of Section 4 and conclude:

Theorem 4.4. *The structural LTL and NBA model-checking problems for DBN-templates are in PSPACE.*

4.2 The special case of restricted DBNs

The difficulty of exponentially long periods is circumvented when we consider restricted DBNs: in this case, the trace is constant from position $t = |\mathbf{V}|^2$ onwards. To show this, we argue that for restricted DBNs, all times $t > |\mathbf{V}|^2$ have the same representative. Since going all the way to $\mathbf{V}^0$ does not give access to any "new" connecting edges in this setting. Formally (see technical appendix for proof):

Claim 2: In a restricted DBN-template, if there is a collision-free path from X^t to Y^t, there is one that goes back at most $|\mathbf{V}|^2$ time slices.

Thus, in the case of restricted DBN, the trace is ultimately constant after at most $|\mathbf{V}|^2$ time steps, and the entire transition system can be computed in time polynomial in $|\mathbf{V}|, |A|$. This reduced complexity allows us to use standard techniques (see technical appendix for precise details) to show:

Theorem 4.5. *The LTL and NBA model-checking problems for restricted DBNs with structural-independence propositions can be solved in polynomial time.*

5 Stochastic conditional independence

In this section, we establish the number-theoretic hardness of reasoning about stochastic CIs when concrete conditional probability distributions are given. Specifically, we shall show that deciding whether formulae of the form $\Diamond(X \perp Y)$ hold is at least as hard as the Skolem problem for rational linear recurrence sequences (LRS).

A rational LRS of order k is a sequence $(u_n)_{n=0}^{\infty}$ of rational numbers satisfying the recurrence relation $u_{n+k} = a_{k-1}u_{n+k-1} + \cdots + a_0 u_n$, where $a_{k-1}, \dots, a_0$ are rational numbers with $a_0 \neq 0$. It is given by the coefficients $a_0, \dots, a_{k-1}$, and the initial terms $u_0, \dots, u_{k-1}$. The Skolem problem takes as input an LRS, where the recurrence relation and initial terms are respectively encoded as a companion matrix $A \in \mathbb{Q}^{k \times k}$ and a vector $u \in \mathbb{Q}^k$, and asks whether there exists an n such that $u_n = 0$, i.e., $A^n u$ contains a 0-entry.

The Skolem problem has been open for nearly a century (Everest et al. 2003; Tao 2008). It is open even if we restrict the LRS to have order five (Ouaknine and Worrell 2012). It is known to be decidable at orders four and below (Tijdeman, Mignotte, and Shorey 1984; Vereshchagin 1985).

Lemma 5.1 (Skolem hardness)**.** *Consider a rational LRS of order k, given by its companion matrix $A \in \mathbb{Q}^{k\times k}$, and vector $u \in \mathbb{Q}^k$ of initial values. We can compute a DBN with $\lceil \log k \rceil + 2$ binary variables $X, Z_1, \ldots, Z_\ell, Y$ where $\ell = \lceil \log k \rceil$ and rational conditional probabilities, such that $\Diamond(X \perp Y)$ holds if and only if the LRS has a zero term.*

The reduction, which we defer to the technical appendix, uses (Aghamov et al. 2025, Cor. 1) to "embed" the given LRS into a Markov chain (M, v), and then lemma 2.4 to convert the Markov chain into a DBN. We remark that as a corollary, our construction can also be used to reduce the closely related Positivity problem for LRS (see, e.g., (Ouaknine and Worrell 2014) for arguments of number-theoretic hardness) to the problem of deciding whether Y always "positively influences" X.

6 Discussion: faithfulness in DBNs

Theorem 3.1 demonstrates an analog of Theorem 2.3 for DBNs, albeit in one direction. However, in future work, we aim to formally prove that the concept of structural independence is faithful to stochastic independence in DBNs, establishing a complete analog of Theorem 2.3 for DBNs. The distinction from the known result is that, when transitioning to DBNs, we impose constraints on the parameters by identifying the distributions of the same variables across different time slices. This reduces the dimensionality of the parameter space, leading to a strictly smaller family of admissible Bayesian networks at any given time t.

We say that the parameters $\langle \mathcal{P}^0, \mathcal{P}^{\mathsf{step}} \rangle$ are *t-unfaithful* if $(\mathbf{X} \perp \mathbf{Y} \mid \mathbf{Z})$ holds at time t but $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ does not. They are called *unfaithful* if this occurs for some t. In other words, unfaithful parameters are those for which the structural and stochastic conditional independencies diverge.

We briefly argue why proving the analog of Theorem 2.3 would suffice. Indeed, recall that by definition, if $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$, then for every $\mathbf{x}, \mathbf{y}, \mathbf{z}$, we must have that the expression

$$\Pr[(\mathbf{X}^t, \mathbf{Y}^t, \mathbf{Z}^t) = (\mathbf{x}, \mathbf{y}, \mathbf{z})] \cdot \Pr[\mathbf{Z}^t = \mathbf{z}] \\ - \Pr[(\mathbf{X}^t, \mathbf{Z}^t) = (\mathbf{x}, \mathbf{z})] \cdot \Pr[(\mathbf{Y}^t, \mathbf{Z}^t) = (\mathbf{y}, \mathbf{z})]$$

is equal to 0. We observe that in the DBN setting, we can use Lemma 2.4 to argue that the above expression is a (degree $O(t)$) polynomial in the parameters $\langle \mathcal{P}^0, \mathcal{P}^{\mathsf{step}} \rangle$. In particular, if the polynomial is not identically 0, then for all but a measure-0 set of parameters, it returns a nonzero value. Proving an analog of Theorem 2.3 would help us deduce that if structural dependence holds, the corresponding polynomial cannot be identically 0.

Using that zero-sets of polynomials have measure 0 and are closed under countable unions, we deduce that unfaithful parameters form a measure-0.

We remark that the proof of Theorem 2.3 in (Meek 1995, Section 6.4) relies on a *local dependence* condition: if (X, Y) is an edge, then X and Y must be dependent. In Bayesian networks, CPDs can be chosen so that variables along a d-path are locally dependent, while all others remain independent of any other variables. This is not possible in dynamic Bayesian networks, where temporal and structural constraints prevent such isolation. This limitation is a key obstacle to extending the argument to the DBN setting. A potential approach is to consider a CPD where a variable takes value 1 with higher probability whenever the majority of its parents are 1. While we focus on binary variables here, establishing the result in this setting would yield the general case as a straightforward corollary. Another promising direction comes from algebraic statistics (**?**), which applies tools from algebraic geometry and combinatorics to study statistical models, especially those involving discrete data. In Bayesian networks, it encodes conditional independencies as polynomial equations and analyzes the resulting algebraic varieties to understand structural and probabilistic properties. Another line of attack could also possibly involve observing that the sequence of polynomials characterizing conditional (in)dependence at time t forms a linear recurrence over the field of multivariate rational functions, and judiciously appealing to the Skolem-Mahler-Lech theorem (the set of zeroes of a linear recurrence over a field of characteristic 0 is the union of a finite set and finitely many effective arithmetic progressions).

7 Conclusion

We introduced LTL-based and NBA-based specification formalisms to express temporal properties of CIs in DBNs. These formalisms can express desirable system properties such as non-interference in security applications and open the possibility to verify systems against all kinds of desirable specifications regarding the temporal evolution of CIs.

We restricted here to CI propositions that state CIs between variables at the same time slice. Our techniques, however, offer the possibility to introduce CI propositions talking about variables at different time slices. A syntax for such propositions could, e.g., be $(\mathbf{X}^{+2} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z}^{+1})$, which holds at time point t in a DBN-template if $(\mathbf{X}^{t+2} \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^{t+1})$. If the entries expressing the time shifts are bounded by some k given in unary, out model-checking algorithm for DBN-templates can be adapted without increasing the asymptotic complexity: We can adapt the construction of the deterministic transition system encoding the trace of the DBN-template by letting the states consist of BN-representatives that unfold the step template for k steps before adding the "tunnelling through"-layer of variables encoding the existence of collision-free d-paths between variables.

Regarding stochastic CIs in DBNs, our Skolem-hardness result is sobering regarding the potential of verifying systems against temporal specifications with respect to stochastic CI statements—which might come as a surprise. The key to obtain this hardness result was establishing the intricate connection between LRSs and DBNs.

References

Aghamov, R.; Baier, C.; Karimov, T.; Nieuwveld, J.; Ouaknine, J.; Piribauer, J.; and Vahanwala, M. 2025. Model Checking Markov Chains as Distribution Transformers. In Jansen, N.; Junges, S.; Kaminski, B. L.; Matheja, C.; Noll, T.; Quatmann, T.; Stoelinga, M.; and Volk, M., eds., *Principles of Verification: Cycling the Probabilistic Landscape : Essays Dedicated to Joost-Pieter Katoen on the Occasion of His 60th Birthday, Part II*, 293–313. Cham: Springer Nature Switzerland. ISBN 978-3-031-75775-4.

Baier, C.; and Katoen, J. 2008. *Principles of model checking*. MIT Press. ISBN 978-0-262-02649-9.

Blondin, M.; and Mckenzie, P. 2014. The Complexity of Intersecting Finite Automata Having Few Final States. In *Computational Complexity*, volume 25. ISBN 978-3-642-30641-9.

Boutilier, C.; Friedman, N.; Goldszmidt, M.; and Koller, D. 1996. Context-Specific Independence in Bayesian Networks. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*, 115–123.

Canonne, C. L.; Diakonikolas, I.; Kane, D. M.; and Stewart, A. 2018. Testing conditional independence of discrete distributions. In Diakonikolas, I.; Kempe, D.; and Henzinger, M., eds., *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, 735–748. ACM.

Everest, G.; van der Poorten, A.; Shparlinski, I.; and Ward, T. 2003. *Recurrence sequences*, volume 104 of *Mathematical Surveys and Monographs*. United States: American Mathematical Society. ISBN 0-8218-3387-1.

Fenton, N.; and Neil, M. 2012. *Risk Assessment and Decision Analysis with Bayesian Networks*. CRC press.

Friedman, N. 2000. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4): 601–620.

Geiger, D.; Verma, T.; and Pearl, J. 1990. Identifying independence in bayesian networks. *Networks*, 20(5): 507–534.

Heckerman, D.; Geiger, D.; and Chickering, D. M. 1995. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning*, 20(3): 197–243.

Jemal, S.; Chaabane, F.; Ben-Hamida, A.; and Borne, P. 2003. Fault diagnosis in complex systems using Bayesian networks. *Control Engineering Practice*, 11(10): 1117–1122.

Klenke, A. 2007. *Probability Theory: A Comprehensive Course*. Springer.

Koller, D.; and Friedman, N. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT press.

Lucas, P.; van der Gaag, L.; and Abu-Hanna, A. 2004. Probabilistic networks in medicine: recent advances and applications. *Artificial Intelligence in Medicine*, 30(3): 201–212.

Manning, C. D.; and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*. MIT press.

Meek, C. 1995. Strong completeness and faithfulness in Bayesian networks. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, 411–418. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN 1558603859.

Meng, G.; Cong, Z.; Li, T.; Wang, C.; Zhou, M.; and Wang, B. 2024. Dynamic Bayesian network structure learning based on an improved bacterial foraging optimization algorithm. *Scientific Reports*, 14: 8266.

Murphy, K. P. 2002. *Dynamic Bayesian networks: Representation, inference and learning*. Ph.D. thesis, University of California, Berkley.

Neapolitan, R. E. 1989. *Probabilistic reasoning in expert systems: theory and algorithms*. Wiley. ISBN 978-0-471-61840-9.

Ouaknine, J.; and Worrell, J. 2012. Decision Problems for Linear Recurrence Sequences. In Finkel, A.; Leroux, J.; and Potapov, I., eds., *Reachability Problems*, 21–28. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-33512-9.

Ouaknine, J.; and Worrell, J. 2014. Positivity problems for low-order linear recurrence sequences. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, 366–379. USA: Society for Industrial and Applied Mathematics. ISBN 9781611973389.

Palaniappan, S. K.; and Thiagarajan, P. S. 2012. Dynamic Bayesian Networks: A Factored Model of Probabilistic Dynamics. In Chakraborty, S.; and Mukund, M., eds., *Automated Technology for Verification and Analysis*, 17–25. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-33386-6.

Pearl, J. 1985. Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning. In *Proceedings of the 7th Conference of the Cognitive Science Society*, 329–334. University of California, Irvine, CA. UCLA Technical Report CSD-850017.

Pearl, J. 1988a. *Probabilistic Reasoning in Intelligent Systems*. San Francisco CA: Morgan Kaufmann. ISBN 978-1-55860-479-7.

Pearl, J. 1988b. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann. ISBN 978-0-08-051489-5.

Pnueli, A. 1977. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, 46–57.

Richard Büchi, J. 1966. Symposium on Decision Problems: On a Decision Method in Restricted Second Order Arithmetic. In Nagel, E.; Suppes, P.; and Tarski, A., eds., *Logic, Methodology and Philosophy of Science*, volume 44 of *Studies in Logic and the Foundations of Mathematics*, 1–11. Elsevier.

Shen, Y.; Huang, H.; Choi, A.; and Darwiche, A. 2019. Conditional Independence in Testing Bayesian Networks. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, 5701–5709. PMLR.

Spohn, W. 1980. Stochastic independence, causal independence, and shieldability. *Journal of Philosophical Logic*, 9(1): 73–99.

Tao, T. 2008. *Structure and Randomness: Pages From Year One of a Mathematical Blog*. American Mathematical Society.

Teymur, O.; and Filippi, S. 2020. A Bayesian nonparametric test for conditional independence. *Foundations of Data Science*, 2(2): 155–172.

Thrun, S.; Burgard, W.; and Fox, D. 2005. *Probabilistic robotics*. MIT press.

Tijdeman, R.; Mignotte, M.; and Shorey, T. 1984. The distance between terms of an algebraic recurrence sequence. *Journal für die reine und angewandte Mathematik*, 349: 63–76.

Vardi, M. Y.; and Wolper, P. 1986. An automata-theoretic approach to automatic program verification. In *1st Symposium in Logic in Computer Science (LICS)*. IEEE Computer Society.

Vereshchagin, N. 1985. The problem of appearance of a zero in a linear recurrence sequence (in Russian). *Mat. Zametki*, 38(2): 609–615.

Wu, J.; Shen, J.; Li, L.; and Xu, G.-Y. 2007. Bayesian networks for object recognition and pose estimation. In *2007 IEEE International Conference on Image Processing*, volume 2, II–285–II–288. IEEE.

Zweig, G.; and Russell, S. 1998. Speech recognition with dynamic Bayesian networks. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence*, AAAI '98/IAAI '98, 173–180. USA: American Association for Artificial Intelligence. ISBN 0262510987.

Reproducibility Checklist

This paper:

- Includes a conceptual outline and/or pseudocode description of AI methods introduced **NA**
- Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results **yes**
- Provides well marked pedagogical references for less-familiare readers to gain background necessary to replicate the paper **yes**

Does this paper make theoretical contributions? **yes**

- All assumptions and restrictions are stated clearly and formally. **yes**
- All novel claims are stated formally (e.g., in theorem statements). **yes**
- Proofs of all novel claims are included. **yes**
- Proof sketches or intuitions are given for complex and/or novel results. **yes**
- Appropriate citations to theoretical tools used are given. **yes**
- All theoretical claims are demonstrated empirically to hold. **NA**
- All experimental code used to eliminate or disprove claims is included. **NA**

Does this paper rely on one or more datasets? **no**

Does this paper include computational experiments? **no**

Technical Appendix to Temporal Properties of Conditional Independence in Dynamic Bayesian Networks

A Probability theory preliminaries

A probability space is a measure space given by a triple $(\Omega, \mathcal{F}, P)$. Here, Ω is the sample space, i.e., a non-empty set of mutually exclusive and collectively exhaustive outcomes (e.g., the set {HH, HT, TH, TT} of outcomes of two coin tosses). The set of events $\mathcal{F}$ is a σ-algebra, i.e., a set of subsets of Ω, called *events*, which includes Ω, and is closed under complement, countable union, and hence also countable intersection by De Morgan's law (in our example we can define $\mathcal{F}$ to have 16 events). Finally, $P : \mathcal{F} \to [0, 1]$ is the probability measure which satisfies $P(\Omega) = 1$ and σ-additivity, i.e., if $\{A_i\}_i$ is a countable collection of pairwise disjoint sets, then $P(\bigcup_i A_i) = \sum_i P(A_i)$ (e.g., $P(\{\}) = 0, P(\{\text{HH}\}) = 1/4, P(\{\text{HT}\}) = 1/4, P(\{\text{TH}\}) = 1/3, P(\{\text{TT}\}) = 1/6$).

A random variable X on a probability space $(\Omega, \mathcal{F}, P)$ is a measurable function from Ω to a measurable space S[2], called its *support*, i.e., for any measurable $T \subseteq S$, $X^{-1}(T) \in \mathcal{F}$. We call X discrete if S is countable, and binary if $S = \{0, 1\}$. In this paper, we shall work with discrete random variables, which will often be binary for simplicity. Given $x \in S$, we write $p_X(x) = \Pr[X = x] = P(\{\omega \in \Omega : X(\omega) = x\})$. For example, for the random variable X which indicates whether the second coin shows Heads, $p_X(1) = 7/12$. We analogously define $p_{\mathbf{X}} : \mathbf{S} \to [0, 1]$ for a finite tuple $\mathbf{X}$ of discrete random variables with support $\mathbf{S}$.

B Proofs of equivalence with Markov chains

We first formally establish the equivalence of DBNs with Markov chains, and then give the reduction from the Skolem problem via its formulation in terms of Markov chains (Aghamov et al. 2025).

Lemma 2.4. *Given a DBN with k binary random variables, we can construct an equivalent Markov chain with 2^k states.*

Given a Markov chain with K states, we can construct an equivalent DBN with $\lceil \log K \rceil$ binary random variables.

Proof. To show the first claim, we describe how to obtain a Markov chain from a DBN: Each of the states of the Markov chain (labelled $0, 1, \dots, 2^{k-1}$) indicates, through the binary expansion of the label, one of the 2^k possible configurations of the random variables. E.g., the state 1 indicates all variables are 0, except $X_0 = 1$.

The conditional probabilities in the step-template network enable us to compute the update matrix M of the Markov chain, column by column (our linear algebraic convention takes the matrix to be column-stochastic, i.e., the columns are distributions and sum up to 1). The (ℓ, m)-th entry is the probability of the configuration being ℓ, given the previous configuration was m.

For the second claim, we show how to encode a Markov chain in a DBN: We assume, without loss of generality, that the Markov chain has $K = 2^{k+1}$ states, labelled $0, 1, \dots, 2^k - 1$. This can be done by adding extra states whose only transitions are self-loops. Our main idea is to perform the previous construction "in reverse."

To that end, we name the required random variables $X_0, \dots, X_k$, and interpret their configuration at the current time slice as indicating the state the Markov chain is in. In the step-template network, each X'_i depends upon $X_0, \dots, X_k, X'_k, \dots, X'_{i+1}$. The idea is the same as that of a binary search: given the previous state of the Markov chain was m (given by $X_k, \dots, X_0$), we first consider the conditional distribution of the most significant bit X'_k of the current state, and then that of the $(k-1)$-st bit X'_{k-1} given $X_k, \dots, X_0, X'_k$, and so on. □

C A faithfulness result

Proposition 3.1. *If $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ in a DBN-template $\mathcal{T}$ for some t, then, $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ in every DBN $\mathcal{B} \in \mathrm{Fam}(\mathcal{T})$.*

Proof. Suppose that $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$. Let $B \in \mathrm{Fam}(\mathcal{T})$ be an arbitrary DBN, and let $B^{0:t}$ denote its unfolding into a BN over time slices 0 to t. Note that $(\mathbf{X}^t \perp\!\!\!\perp \mathbf{Y}^t \mid \mathbf{Z}^t)$ holds iff there is no d-path wrt $\mathbf{Z}^t$ from $\mathbf{X}^t$ to $\mathbf{Y}^t$ in the BN-template $\mathcal{T}^{0:t}$. By Theorem 2.3, it follows that $(\mathbf{X}^t \perp \mathbf{Y}^t \mid \mathbf{Z}^t)$. □

D Structural independence-related proofs

Lemma 4.3. *The LTL and NBA model-checking problems with structural-independence propositions are hard for NP as well as for coNP.*

Proof. Assume we are given unary DFAs $\mathcal{A}_1, \dots, \mathcal{A}_k$. Recall that each $\mathcal{A}_i$ is given by a set $\mathbf{Q}_i = \{Q_{i,0}, \dots, Q_{i,m_i}\}$ of states, unary alphabet $\{a\}$, the initial state $Q_{i,0}$, transition function $\delta_i : \mathbf{Q}_i \to \mathbf{Q}_i$, and set of accepting states $\mathbf{F}_i \subseteq \mathbf{Q}_i$. The intersection problem asks whether there exists $t \geq 0$ such that a^t is accepted by all the given automata. We encode this as a DBN with variables $\{X_0, \dots, X_k\} \cup \mathbf{Q}_1 \cdots \cup \mathbf{Q}_k$. Let $\mathbf{F} = \mathbf{F}_1 \cup \dots \cup \mathbf{F}_k$. In the initial template, we build bridges $(X^0_{i-1}, Q^0_{i,0}), (X^0_i, Q^0_{i,0})$. In the step template, we connect $(X_0, X'_0), (X_k, X'_k)$, and $(Q_{i,j}, \delta_i(Q_{i,j})')$.

We then observe that a^t is accepted by all automata if and only if $(X_0 \perp\!\!\!\perp X_k \mid \mathbf{F})$ holds at time t. To see this, note that a d-path from X^t_0 to X^t_k first has to go up to X^0_0. Then, it has to move to $Q^0_{1,0}$ from there, it can only continue to X^0_1 if $Q^0_{1,0}$ (or one of its decendants) can serve as a collider. This is only possible if an element of $\mathbf{F}^t$ is reachable from $Q^0_{1,0}$, which is the case if and only if the word a^t leads to an accepting state in $\mathcal{A}_1$. Analogously, a^t also has to be accepted by all other given unary DFAs. □

Claim 1: If there exists a d-path from X to Y with respect to a set $\mathbf{Z}$ of size k in an arbitrary BN, there is one with at most k collisions, all of which occur inside $\mathbf{Z}$.

Proof. Suppose, for the sake of contradiction, a d-path with the fewest collisions has more than k collisions. Then, by the pigeonhole principle there exists $Z \in \mathbf{Z}$ to which more than one collision is attributed. Let the first and last of these collisions occur at W_1, W_2 respectively. Since they have Z

[2] We equip S with σ-algebra $\mathcal{F}'$ and measure $\mu : \mathcal{F}' \to \mathbb{R}$.

as a common ancestor, there exists a path from W_1 to W_2 with a single collision inside $\mathbf{Z}$. We use this path to "tunnel through" and replace the given d-path with one that has fewer collisions: a contradiction, as desired. □

Claim 2: In a restricted DBN-template, if there is a collision-free path from X^t to Y^t, there is one that goes back at most $|\mathbf{V}|^2$ time slices.

Proof. Suppose, for the sake of contradiction, a path with the lowest "traceback" goes back more than $|\mathbf{V}|^2$ time slices. Then, by the pigeonhole principle, there exist $i < j$ such that the path enters and exits $\mathbf{V}^{t-i}$ and $\mathbf{V}^{t-j}$ at the same pair of variables. We can thus replace the path from $\mathbf{V}^{t-i}$ with the path from $\mathbf{V}^{t-j}$ and get a collision-free path with a smaller traceback: a contradiction, as desired. □

Theorem 4.5. *The LTL and NBA model-checking problems for restricted DBNs with structural-independence propositions can be solved in polynomial time.*

Proof. The case of NBA model checking is immediate. We can construct an automaton whose language is precisely the trace τ, intersect it with the given NBA, and check the result for non-emptiness, all in polynomial time (Baier and Katoen 2008, Chapter 4.3).

To check whether an ultimately constant trace satisfies an LTL formula φ, we adopt a dynamic programming approach, memoizing for $0 \le t \le |\mathbf{V}|^2 + 1 = T$ whether a suffix $\tau[t : \infty]$ satisfies a subformula φ' of φ. We populate entries from larger to smaller t, and simpler to more complex φ. Atomic propositions and Boolean connectives are handled in the obvious way. The suffix from time $t = T$ satisfies $\bigcirc\varphi'$ if and only if it satisfies φ'; it satisfies $\varphi_1 \mathcal{U} \varphi_2$ if and only if it satisfies φ_2. For smaller t, the suffix from time t satisfies $\bigcirc\varphi'$ if and only if the suffix from time $t+1$ satisfies φ; it satisfies $\varphi_1 \mathcal{U} \varphi_2$ if and only if it either satisfies φ_2, or it satisfies φ_1 and the suffix from $t+1$ satisfies $\varphi_1 \mathcal{U} \varphi_2$. Having populated the table, we check whether the suffix from $t = 0$ satisfies the given φ. □

E Proof of Skolem-hardness

Lemma 5.1 (Skolem hardness)**.** *Consider a rational LRS of order k, given by its companion matrix $A \in \mathbb{Q}^{k\times k}$, and vector $u \in \mathbb{Q}^k$ of initial values. We can compute a DBN with $\lceil \log k \rceil + 2$ binary variables $X, Z_1, \dots, Z_\ell, Y$ where $\ell = \lceil \log k \rceil$ and rational conditional probabilities, such that $\Diamond(X \perp Y)$ holds if and only if the LRS has a zero term.*

Proof. We construct a DBN such that at the n-th time step:

- The event $Y = 1$ occurs unconditionally with probability $1/2$.
- The difference $\Pr[X = 1|Y = 1] - \Pr[X = 1|Y = 0]$ is $2^{-n}\rho^n\eta$ times the n-th term of the LRS, where η, ρ are positive rational constants.

At a high level, the construction proceeds as follows:

1. We use (Aghamov et al. 2025, Cor. 1) to "embed" the given instance (A, u) of order k into an ergodic Markov chain (M, v) of order $k+1$.
2. We encode the states of the Markov chain with binary variables, or "bits" $X, Z_0, \dots, Z_\ell$, where X indicates whether the system is in the first state, and all other states are indicated by the usual binary encoding introduced in Lemma 2.4. This ensures that using even restricted DBNs suffices for the reduction.
3. In the DBN, the current values of $X, Z_0, \dots, Z_\ell$ depend on not only on the previous values, but also on the current value of Y. If $Y = 1$, the distribution of current values is obtained from the previous values as per the construction in Lem. 2.4. Otherwise, the distribution is "fast-forwarded" to the stationary distribution s of M.

We remark that as a corollary, our construction can also be used to reduce the closely related Positivity problem for LRS (see, e.g., (Ouaknine and Worrell 2014) for arguments of number-theoretic hardness) to the problem of deciding whether Y always "positively influences" X. Let $s \in \mathbb{Q}^{k+1}$ be the uniform distribution, and S be the square matrix whose columns are all s. By (Aghamov et al. 2025, Cor. 1), we can compute an ergodic Markov chain M and an initial distribution v (both with all entries rational) such that for all n,

$$M^n v = s + \eta\rho^n \begin{bmatrix} I \\ -\mathbf{1}_k^\top \end{bmatrix} A^n u, \tag{1}$$

where η, ρ are positive rational constants, and $\mathbf{1}_k$ denotes the vector with all entries equal to 1.

We label the states of the Markov chain as $\alpha, 0, 1, \dots, k-1$, and encode them with bits $X, Z_0, \dots, Z_\ell$. The encoding of α has $X = 1$ and all other bits 0, the encoding of any other state i has $X = 0$ and other bits set according to the binary representation of i. Clearly, this uses $\lceil \log k \rceil + 1$ bits.

We shall also index the rows and columns of M by $\alpha, 0, 1, \dots, k-1$, such that α corresponds to the topmost, and leftmost.

To construct the DBN including the additional bit c, we replicate the construction of Lem. 2.4. The initial Bayesian network is set up so that valid state encodings with $Y = 0$ each get half the probability prescribed by s, and valid state encodings with $Y = 1$ each get half the probability prescribed by v.

For the step-template, we have that the current value of Y is unconditionally assigned uniformly at random. If $Y = 1$, the conditional distribution of bits $X, Z_0, \dots, Z_\ell$ is the same as that prescribed by M, i.e., if the previous values encoded a valid state i, then the conditional probability of valid encoding j is the (j, i)-th entry of M; if the previous value was an invalid encoding i, then the current value is deterministically i. Similarly, if $Y = 0$: if the previous values encoded a valid state i, then all valid encodings j are assigned probability $1/(k+1)$, if the previous encoding i was invalid, then the current encoding is deterministically i.

We note that only the valid encodings are reachable, and $\frac{1}{2}\begin{bmatrix} M & M \\ S & S \end{bmatrix}$ is an equivalent Markov chain for (the reachable configurations of) the DBN, and the initial distribution is $\frac{1}{2}\begin{bmatrix} v \\ s \end{bmatrix}$.

Intuitively, at step n, the probability that $X, Z_1, \dots, Z_\ell$ encode state i is the probability that there was never a fast-forward (which is 2^{-n-1}) times what the probability would be according to the Markov chain (which is $e_i^\top M^n v$), plus the probability there was a fast-forward times the stationary probability (which is $e_i^\top s$), where e_i is the vector whose entry at index i is 1 and other entries are 0.

Formally, at step n,

$\Pr[X = 1] = \frac{1}{2^{n+1}} e_\alpha^\top M^n v + \frac{2^{n+1}-1}{2^{n+1}} \cdot \frac{1}{k+1}$.

We can also check this via the equivalent Markov chain of the DBN. We can show via a simple induction, and using the facts that $MS = SM = SS = S$, that

$$\begin{bmatrix} M & M \\ S & S \end{bmatrix}^n = \begin{bmatrix} M^n + (2^{n-1} - 1)S & M^n + (2^{n-1} - 1)S \\ 2^{n-1}S & 2^{n-1}S \end{bmatrix}$$

Using the fact that $Sv = s$, we have that at step n, the distribution is $\frac{1}{2^{n+1}} \begin{bmatrix} M^n v + (2^n - 1)s \\ 2^n s \end{bmatrix}$.

Observe that $\Pr[X = 1 \mid Y = 1] = \frac{1}{2^n} e_\alpha^\top M^n v + \frac{2^n - 1}{2^n} \cdot \frac{1}{k+1}$, and that $\Pr[X = 1 \mid Y = 0] = \frac{1}{k+1}$, and their difference is $\frac{1}{2^n} \left(e_\alpha^\top M^n v - \frac{1}{k+1} \right)$. Recall that by the embedding 1 of the LRS by (Aghamov et al. 2025, Cor. 1), this can be rewritten as $2^{-n} \eta \rho^n (e^\top A^n u)$, i.e., a scaled version of the LRS. We have thus shown that at time n, $\Pr[X = 1 \mid Y = 1] - \Pr[X = 1 \mid Y = 0] = 2^{-n} \eta \rho^n u(n)$, where $u(n)$ is the n-th term of the LRS, and η, ρ are rational constants. The reductions of Skolem to eventual independence of X and Y and Positivity to global causation of X by Y is complete. $\square$