Convergence in Markovian Models with Implications for Efficiency of Inference ⋆

Theodore Charitos *, Peter R. de Waal , Linda C. van der Gaag

Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands

Abstract

Sequential statistical models such as dynamic Bayesian networks and hidden Markov models more specifically, model stochastic processes over time. In this paper, we study for these models the effect of consecutive similar observations on the posterior probability distribution of the represented process. We show that, given such observations, the posterior distribution converges to a limit distribution. Building upon the rate of the convergence, we further show that, given some wished-for level of accuracy, part of the inference can be forestalled. To evaluate our theoretical results, we study their implications for a real-life model from the medical domain and for a benchmark model for agricultural purposes. Our results indicate that whenever consecutive similar observations arise, the computational requirements of inference in Markovian models can be drastically reduced.

Key words: Markovian models; consecutive similar observations; convergence; inference; efficiency

1 Introduction

Sequential statistical models for reasoning about stochastic processes include hidden Markov models (HMMs) and dynamic Bayesian networks (DBNs); when these models satisfy the *Markovian property*, where the future state of the represented process is assumed to be independent of the past state given its present state, we call them *Markovian*. Markovian models represent the dynamics of a discrete-time

⋆ This research was (partly) supported by the Netherlands Organisation for Scientific Research (NWO).

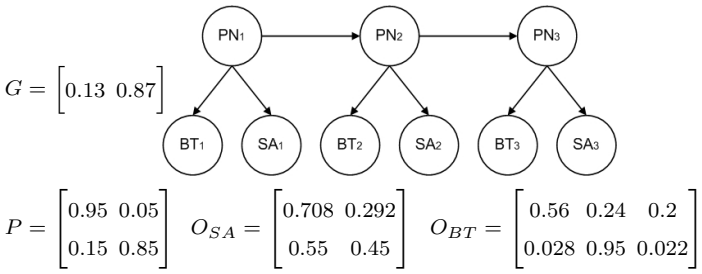
* Corresponding author. Tel.: +31 30 253 9827; fax: +31 30 251 3791.
Email addresses: theodore@cs.uu.nl (Theodore Charitos), waal@cs.uu.nl (Peter R. de Waal), linda@cs.uu.nl (Linda C. van der Gaag).

process by explicitly specifying a stochastic transition rule for the change of the state of the process over time. DBNs [9,13,16] model the interactions among a collection of dynamic variables and in essence constitute an extension of HMMs which capture the dynamics of a single variable [3,15,19]. Applications of Markovian models include medical diagnosis [1,5] and treatment planning [18], speech recognition [3,19], computational biology [6], and reliability engineering [23].

Exact inference in Markovian models is computationally hard, since the runtime requirements of the available algorithms are exponential in the number of variables that represent the unknown hidden state [4,16]. In this paper, we will show that the nature of the observations obtained may help reduce the requirements involved. We will show more specifically that, after a specific number of consecutive similar observations have been propagated, the posterior distribution of the stochastic process has converged to a limit distribution within some level of accuracy. Continuing to obtain similar observations will not alter the distribution beyond this level and therefore no further inference is required. The total number of time slices over which we need to perform inference can thus be drastically reduced, leading to considerable computational savings. The achieved reduction depends upon the wished-for level of accuracy: the higher the accuracy we want, the fewer the savings will be. It is well known from the literature on Markov chains [12,20], that an ergodic Markov chain converges geometrically to a stationary distribution that is (in the limit) independent of the initial distribution of its states. To the best of our knowledge, integration of these results into Markovian models in general with the aim of reducing the computational requirements involved has not been addressed before.

In this paper we initially restrict our presentation to HMMs, using an example application from the medical domain. We subsequently indicate how our method can be extended to Markovian models with a richer structure in their set of observable variables and to models that capture interventions of the modelled process. We further show how our analysis applies to Markovian models consisting of interacting processes. We validate our theoretical results on the dVAP model for the diagnosis of pneumonia in mechanically ventilated patients [5] and on the Mildew model for forecasting mildew fungus and gross yield from a field wheat [13]. Our experimental results support our theoretical analysis and show that consecutive similar observations can play a significant role in speeding up inference. For some patients in the dVAP model especially, we achieved a reduction of the number of computations involved by a factor of the order of 2^{13}, which led to a substantial speed up of the inference.

The remainder of the paper is organised as follows. In Section 2, we set out by introducing the real-life application that motivated our study. In Section 3 we discuss inference in Markovian models and propose an alternative framework for inference with HMMs that is tailored to our analysis. We continue in Section 4 by studying the effect of consecutive similar observations in HMMs and determining the con-



$$G = \begin{bmatrix} 0.13 & 0.87 \end{bmatrix} \quad P = \begin{bmatrix} 0.95 & 0.05 \\ 0.15 & 0.85 \end{bmatrix} \quad O_{SA} = \begin{bmatrix} 0.708 & 0.292 \\ 0.55 & 0.45 \end{bmatrix} \quad O_{BT} = \begin{bmatrix} 0.56 & 0.24 & 0.2 \\ 0.028 & 0.95 & 0.022 \end{bmatrix}$$

Fig. 1. A dynamic model for the evolution of pneumonia with two observable variables; the probability tables are obtained from [21].

vergence rate for the probability distribution of the hidden process. We then address in Section 5 the effect of consecutive similar observations for Markovian models with richer structure. In Section 6, we analyse the runtime savings that are achieved by forestalling part of the inference and illustrate these savings on the dVAP and Mildew models. We end the paper with our conclusions in Section 7.

2 A motivating example

Throughout the paper we will use the dynamic model from Figure 1 for our running example. The model constitutes a fragment of a temporal Bayesian network that was developed for the management of ventilator associated pneumonia (VAP) in patients at an Intensive Care Unit (ICU) [14,21]. Pneumonia, denoted as *PN*, constitutes the binary unobservable variable that we would like to study over time. The observable variables model a patient's body temperature, denoted as *BT*, with values $\{> 38.5\,^{\circ}\mathrm{C},\ \text{normal}, < 36.0\,^{\circ}\mathrm{C}\}$, and sputum amount, denoted as *SA*, with values $\{\text{yes}, \text{no}\}$. The observable variables are measured every two hours. As an example, Figures 2 and 3 illustrate the data obtained for two patients on a specific day. From Figure 2 we note that within the data for patient Id.1051, two sequences of consecutive similar observations can be discerned per variable; for both variables combined, three such sequences are found. From Figure 3 pertaining to patient Id.851, two sequences of consecutive similar observations can be discerned for *BT* and three sequences for *SA*; for both variables combined, there also are three sequences. Table 1 summarises these findings.

We now are interested in determining whether we need to use all the data that are available for a particular patient to establish the probability distribution of the variable *PN* within reasonable accuracy. For example, using the model from Figure 1, we compute the probability of pneumonia at time 22:00 for patient Id.1051 to be 0.997887. This probability does not differ much from the probability at time 20:00 which is 0.997881, nor from that at time 18:00 which is 0.997726. Similarly for patient Id.851 we find the probability of pneumonia at time 22:00 to be 0.034551, while at time 20:00 this probability is 0.036490 and at time 18:00 it is 0.047090. Since after a specific number of consecutive similar observations the probability

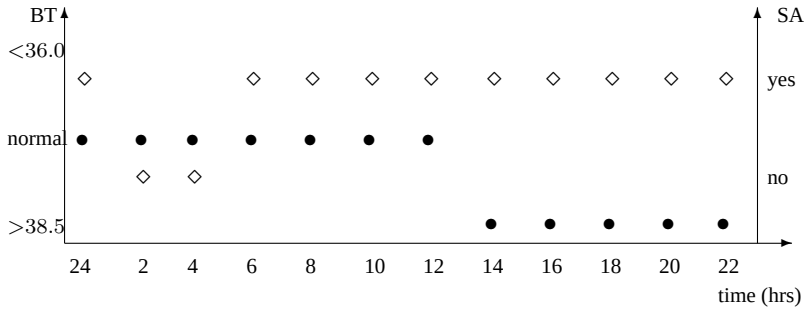


Fig. 2. The data for patient Id.1015 on a specific day, where • is used for a BT observation and ◇ for an SA observation.

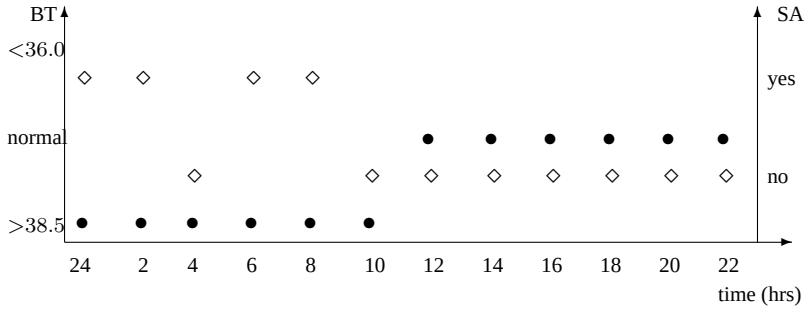


Fig. 3. The data for patient Id.851 on a specific day, where • is used for a BT observation and ◇ for an SA observation.

distribution of the hidden process does not change much with respect to a given level of accuracy, it is worthwhile to investigate whether we can forestall part of the inference.

3 Markovian models

We review some basic concepts from the theory of Markovian models [15,16,19], and present an alternative framework for inference with HMMs that is tailored to our analysis.

3.1 Basic notions

An HMM can be looked upon as an extension of a finite Markov chain, by including observable variables that depend on the hidden variable. We use X_n to denote the hidden variable at time n, with states $S_X = \{1, 2, \ldots, m\}$, $m \geq 1$. We denote the prior probability distribution of the hidden variable at time 1 by the vector G, with probabilities $g_i = p(X_1 = i)$. The transition behaviour of a Markov chain is generally represented by a matrix P of *transition probabilities*. We consider only homogeneous Markov chains in which the transition probabilities do not depend on time, and define $p_{ij} = p(X_{n+1} = j \mid X_n = i)$ for every $i, j = 1, \ldots, m, n \geq 1$. The transition matrix P from Figure 1, for example, indicates that if a patient does not

Observations	**Id.1051**	**Id.851**
BT=normal	24:00-12:00	12:00-22:00
BT=> 38.5	14:00-22:00	24:00-10:00
SA=no	2:00-4:00	10:00-22:00
SA=yes	6:00-22:00	24:00-2:00,6:00-8:00

Set of observations	**Id.1051**	**Id.851**
BT=normal, SA=no	2:00-4:00	12:00-22:00
BT=normal, SA=yes	6:00-12:00	-
BT=> 38.5, SA=yes	14:00-22:00	24:00-2:00,6:00-8:00

Table 1
The sequences of consecutive similar observations per variable and for both variables combined, from the data for patients Id.1015 and Id.851.

have pneumonia at a particular time n, then there is a probability of 0.15 that she/he will have developed pneumonia at time $n + 1$. We assume that the diagonal of the transition matrix has non-zero elements only, that is, we assume that it is possible for each state to persist. We denote the observable variables by Y_n, with values $S_Y = \{1, 2, \ldots, r\}, r \geq 1$. The observations are generated from the state of the hidden variable according to a time-invariant probability distribution matrix O, where the (i, j)-th entry gives, for each time $n \geq 1$, the probability of observing $Y_n = j$ given that the hidden variable X_n is in state i, that is, $o_{ij} = p(Y_n = j \mid X_n = i)$. The observation matrix O_{BT} from Figure 1, for example, states that the probability that a patient will show a high temperature given that she/he has pneumonia, is 0.56; if she/he does not have pneumonia, this probability is just 0.028.

A DBN can be looked upon as an extension of an HMM, that captures a process that involves a collection of hidden variables. The set of variables $\mathbf{V}_n$ of the model is partitioned into three mutually exclusive and collectively exhaustive sets $\mathbf{I}_n, \mathbf{X}_n, \mathbf{Y}_n$, where the sets $\mathbf{I}_n$ and $\mathbf{Y}_n$ constitute the input and output variables at time n respectively, and $\mathbf{X}_n$ includes the hidden variables. The joint probability distribution over the variables at a particular time is captured in a factorised way by a graphical model.

3.2 Inference in Markovian models

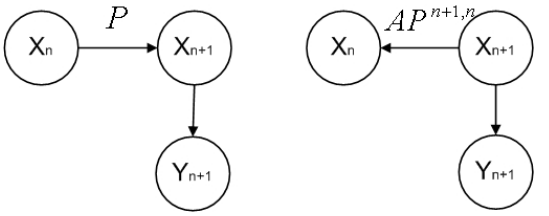
When applying Markovian models, usually the probability distributions of the hidden variables are computed using an inference algorithm. Three different types of inference are distinguished, which are monitoring, smoothing and forecasting. *Monitoring* is the task of computing the probability distributions for $\mathbf{X}_n$ at time n given observations that are available up to and including time n. *Smoothing* (or *diagnosis*) is the task of computing the probability distributions for $\mathbf{X}_n$ at time n given observations from the future up to time N, where $N > n$. Finally, *forecasting* is the task of predicting the probability distributions of $\mathbf{X}_n$ at time n given observations about the past up to and including time N, where $N < n$.

For exact inference with an HMM, an efficient recursive scheme, called the Forward-Backward, or Baum-Welch, algorithm, has been proposed [2]; this algorithm was introduced originally for finding unknown parameter probabilities for an HMM, but can also be used for inference purposes [19]. Instead of using the Forward-Backward algorithm directly, we propose an alternative framework for inference with HMMs that is better suited to our analysis of the effect of consecutive similar observations and is directly related to the concepts from linear algebra that we will use in later sections. Our framework uses an explicit representation of the matrix multiplications that are involved in inference. It further builds upon the concept of arc reversal for smoothing [22]. Our framework can in addition be readily extended to Markovian models with conditionally independent observable variables as we will show in Section 5.

We denote by D_N the set of observations that are available up to and including time N; we assume that there are no missing values in D_N. We further denote by $OM(j) = diag(O_{1j}, \ldots, O_{mj})$, $j = 1, \ldots, r$, the diagonal matrix that is constructed from the jth column of the observation matrix O; we call this matrix the *observation column* matrix for observation j. The *present row vector* PV_n for time n now is defined as $PV_n(i) = p(X_n = i \mid D_n)$, $i = 1, \ldots, m$, and is computed recursively as follows:

- at time 1, if there is an observation j, we take $PV_1 = G \cdot OM(j)$;
- at time $n = 2, \ldots, N$, if there is an observation j, we take $PV_n = PV_{n-1} \cdot P \cdot OM(j)$.

In each step, we normalise the vector PV_n by dividing it by $\sum_{i=1}^{m} PV_n(i)$. As an example, we consider a patient who has a normal temperature for the first two time slices. We are interested in the probability that this patient has pneumonia at time



Forward influence Backward influence

Fig. 4. Arc reversal in an HMM.

2. For time 1, we compute the present row vector PV_1 to be

$$PV_1 = \begin{bmatrix} 0.13 & 0.87 \end{bmatrix} \cdot \begin{bmatrix} 0.24 & 0 \\ 0 & 0.95 \end{bmatrix} = \begin{bmatrix} 0.0364 & 0.9636 \end{bmatrix}$$

For time 2, we find for the present row vector that

$$PV_2 = \begin{bmatrix} 0.0364 & 0.9636 \end{bmatrix} \cdot \begin{bmatrix} 0.95 & 0.05 \\ 0.15 & 0.85 \end{bmatrix} \cdot \begin{bmatrix} 0.24 & 0 \\ 0 & 0.95 \end{bmatrix} = \ldots = \begin{bmatrix} 0.0522 & 0.9478 \end{bmatrix}$$

The probability that this patient currently has pneumonia therefore is just 0.0522.

For forecasting the probability distribution of the hidden variable X_n at some time $n > N$ in the future, we define the *future row vector* $FV_{n,N}$ by $FV_{n,N}(i) = p(X_n = i \mid D_N)$, $i = 1, \ldots, m$. The vector is computed as

$$FV_{n,N} = PV_N \cdot P^{n-N} \tag{1}$$

For computing a smoothed probability distribution for some time $n < N$ in the past, we define the *backward row vector* $BV_{n,N}$ by $BV_{n,N}(i) = p(X_n = i \mid D_N), i = 1, \ldots, m$. The backward row vector can be computed recursively by applying evidence absorption and arc reversal [22]; Figure 4 illustrates the basic idea. We observe that the states of the variable X_n affect the probability distribution of the variable X_{n+1} via the transition matrix P. By using Bayes' theorem

$$p(X_n \mid X_{n+1}) = \frac{p(X_{n+1} \mid X_n) \cdot p(X_n)}{p(X_{n+1})} \tag{2}$$

we find that the states of the variable X_{n+1} affect the probability distribution of the variable X_n via the matrix $AP^{n+1,n}$, with $AP^{n+1,n}(ij) = p(X_n = j \mid X_{n+1} = i), i, j = 1, \ldots, m$. The matrix $AP^{n+1,n}$ is established for $n = 1, \ldots, N-1$ from

- $p(X_n) = PV_n$;
- $p(X_{n+1}) = p(X_n) \cdot P$;
- $AP^{n+1,n}$ is computed from $p(X_n \mid X_{n+1})$ using equation (2).

The backward row vector $BV_{n,N}$ then is computed recursively from

- $BV_{N,N} = PV_N$;
- for $n = N-1, \ldots, 1$, we take $BV_{n,N} = BV_{n+1,N} \cdot AP^{n+1,n}$.

Again, we normalise the vector $BV_{n,N}$ in each step by dividing by $\sum_{i=1}^{m} BV_{n,N}(i)$. Note that the matrix $AP^{n+1,n}$ essentially represents the reversed transition behaviour of the process and determines the strength of the influence of observations in the future on the probability distribution of the hidden process in previous times. Also note that the matrix $AP^{n+1,n}$ is well-defined when each state of the represented process has a non-zero probability of persisting.

In essence, the computational complexity of our framework is the same as that of the Forward-Backward algorithm when used for inference [15,16]. An extension of our inference framework to Markovian models with multiple interacting subprocesses is possible. Since the number of states of the overall process grows exponentially with the number of subprocesses, however, maintaining an explicit representation of a transition and an observation matrix will be infeasible. A more efficient algorithm then is the *interface algorithm* [16]. This algorithm is an extension of the *junction-tree algorithm* for inference in Bayesian networks in general [8]. It efficiently exploits the concept of *forward interface*, which is the set of variables at time n that affect some variables at time $n+1$ directly. The complexity of the interface algorithm has been shown to lie between $\Omega(M^{I+1})$ and $O(M^{I+D})$, where I is the size of the forward interface, D is the number of hidden variables, and M is the maximum number of values that a hidden variable in the model can take. We show in later sections, both theoretically and experimentally, that the nature of the observations obtained and the graphical structure of the Markovian model can be exploited to effectively reduce the runtime requirements of the interface algorithm.

4 Consecutive similar observations

We analyse the effect of observing consecutive similar values for an observable variable on the probability distribution of the hidden variable. More specifically, we are interested in the convergence behaviour of the posterior distribution of the variable X_n in terms of the number k_j of consecutive observations j. We will argue that, given a specific k_j, observing more similar values will not alter the probability distribution of the hidden variable beyond a given level of accuracy.

We consider an HMM with a single observable variable and an associated dataset D_N. We suppose that the same value j is observed for this variable from time n up to and including time N for some $n < N$; the number of consecutive similar observations thus is $k_j = N - (n-1)$. Using our inference framework, the present row vector PV_N is computed to be

$$\begin{aligned} PV_N &= \alpha(k_j, PV_{n-1}) \cdot PV_{n-1} \cdot (P \cdot OM(j))^{k_j} \\ &= \alpha(k_j, PV_{n-1}) \cdot PV_{n-1} \cdot (R_j)^{k_j} \end{aligned} \tag{3}$$

where $\alpha(k_j, PV_{n-1})$ is a normalisation constant that depends on k_j and PV_{n-1}, and R_j is the square matrix $R_j = P \cdot OM(j)$. We will now use equation (3) to study the convergence of the present row vector to a limit distribution. More specifically, we would like to estimate the number k_j of consecutive similar observations such that

$$|PV_{k_j+1} - PV_{k_j}|_\infty \leq \theta$$

where $\theta > 0$ is a predefined level of accuracy and $|w|_\infty \equiv \max_i |w_i|$ denotes the L^∞ norm of a vector $w = (w_1, \ldots, w_m)$. We then have that observing more than k_j consecutive similar values will add no extra information to the probability distribution of the hidden variable and no further inference needs to be performed. To establish the convergence behaviour of the present row vector and of the matrix R_j more specifically, we build upon the notion of *spectral radius*, where the *spectral radius* $\rho(A)$ of a square matrix A is defined as $\rho(A) \equiv \max\{|\lambda| : \lambda \text{ is an eigenvalue of } A\}$. The following theorem [11, Theorem 5.6.12] reviews a necessary and sufficient condition for the convergence of reflexive multiplication of a square matrix in terms of its spectral radius.

Theorem 1 *Let A be a square matrix. Then,* $\lim_{k\to\infty} A^k = 0$ *if and only if* $\rho(A) < 1$.

To study the spectral radius of the matrix R_j, we recall that R_j is the product of a stochastic matrix P and the nonnegative diagonal observation column matrix $OM(j)$. The following proposition now states a property of the spectral radius of such a product, based upon which we will argue that $\rho(R_j) < 1$ for any non-trivial R_j.

Proposition 1 *Let A be a stochastic matrix and let B be a diagonal matrix. Then,* $\rho(A \cdot B) \leq \rho(B)$.

Proof. From [11, Theorem 5.6.9] we have that for any square matrix A, it holds that $\rho(A) \leq \|A\|$, where $\|.\|$ is any matrix norm. From this property we have that $\rho(A \cdot B) \leq \|A \cdot B\|$. Now, any matrix norm satisfies the submultiplicative axiom which states that $\|A \cdot B\| \leq \|A\| \cdot \|B\|$. Hence, $\rho(A \cdot B) \leq \|A\| \cdot \|B\|$. By choosing the *maximum row sum matrix* norm $\|.\|_\infty$ which is defined on A as

$$\|A\|_\infty \equiv \max_i \sum_{j=1}^{n} |a_{ij}|$$

we find that $\|A\|_\infty = \rho(A) = 1$ and $\|B\|_\infty = \rho(B)$. The property stated in the proposition now follows directly. □

From Proposition 1 we conclude for the spectral radius of the matrix R_j that $\rho(R_j) \leq \rho(OM(j)) \leq 1$. We note that $\rho(R_j) = 1$ only if $OM(j)$ is the identity matrix, which

basically means that the observation j is deterministically related with the hidden state, or the hidden process itself is deterministic and at least one element of $OM(j)$ equals one. For any non-trivial transition matrix and observation column matrix, therefore, we have that $\rho(R_j) < 1$. From Theorem 1 we can now conclude that $\lim_{k_j \to \infty} R_j^{k_j} = 0$. Note that from this property we cannot yet conclude that the present row vector PV_N converges to some limit distribution, since we also need to establish the limit behaviour of the normalisation constant $\alpha(k_j, PV_{n-1})$. We recall that the normalisation constant is dependent not only of k_j but of PV_{n-1} as well. If PV_N converges, it will converge to a probability distribution, which implies that $\alpha(k_j, PV_{n-1})$ will diverge according to $1/\rho(R_j)^{k_j}$. To establish whether or not PV_N converges therefore, we have to look at the limit behaviour of $\alpha(k_j, PV_{n-1}) \cdot (R_j)^{k_j}$. For this purpose, we build upon the following theorem, known as Perron's theorem [11, Theorem 8.2.11], which provides a limit matrix for $[\rho(R_j)^{-1} \cdot R_j]^{k_j}$.

Theorem 2 *(Perron's theorem) Let A be a square matrix with positive elements. Then,* $\lim_{k \to \infty} [\rho(A)^{-1} \cdot A]^k = L_A$ *where* $L_A \equiv x \cdot y^T$, *with* $A \cdot x = \rho(A) \cdot x,\ A^T \cdot y = \rho(A) \cdot y,\ x > 0, y > 0,$ *and* $x^T \cdot y = 1$.

Before applying Perron's theorem to $[\rho(R_j)^{-1} \cdot R_j]^{k_j}$, we re-arrange equation (3) to get

$$PV_N = PV_{n-1} \cdot \alpha(k_j, PV_{n-1}) \cdot \rho(R_j)^{k_j} \cdot (R_j/\rho(R_j))^{k_j} \tag{4}$$

Note that $\alpha(k_j, PV_{n-1})$ being a non-linear function of PV_{n-1} prohibits a straightforward application of Perron's theorem in equation (4). We therefore begin by showing that we can indeed establish the convergence of PV_N by building upon the theorem of Perron.

Proposition 2 *Let* $c_{k_j} \equiv \alpha_{k_j}(PV_{n-1}) \cdot \rho(R_j)^{k_j}$, *where* $\alpha_{k_j}(PV_{n-1}), k_j$ *and* R_j *are as in equation (3). Then,* $\lim_{k_j \to \infty} c_{k_j} = c$ *for some constant* $c > 0$, *and* $\lim_{k_j \to \infty} PV_N = c \cdot PV_{n-1} \cdot L_{R_j}$, *where* L_{R_j} *is as defined in Theorem 2.*

Proof. By definition we have that

$$c_{k_j} = \alpha(k_j, PV_{n-1}) \cdot \rho(R_j)^{k_j} = \frac{\rho(R_j)^{k_j}}{\sum_i (PV_{n-1} \cdot R_j^{k_j})(i)}$$

From Theorem 2, we now find that $\lim_{k_j \to \infty} c_{k_j} = c$, where c equals

$$c = \left[\sum_i (PV_{n-1} \cdot L_{R_j})(i) \right]^{-1} > 0$$

For any vector norm we further have that

$$\begin{aligned}
&\left\|\alpha(k_j, PV_{n-1})\cdot PV_{n-1}\cdot R_j^{k_j} - c\cdot PV_{n-1}\cdot L_{R_j}\right\| \\
=&\left\|\alpha(k_j, PV_{n-1})\cdot PV_{n-1}\cdot \rho(R_j)^{k_j}\cdot\left[\tfrac{R_j}{\rho(R_j)}\right]^{k_j} - c\cdot PV_{n-1}\cdot L_{R_j}\right\| \\
=&\left\|c_{k_j}\cdot PV_{n-1}\cdot\left[\tfrac{R_j}{\rho(R_j)}\right]^{k_j} - c\cdot PV_{n-1}\cdot L_{R_j}\right\| \\
\leq&|c_{k_j}-c|\cdot\left\|PV_{n-1}\cdot\tfrac{R_j^{k_j}}{\rho(R_j)^{k_j}}\right\| + \left\|c\cdot PV_{n-1}\right\|\cdot\left\|\tfrac{R_j^{k_j}}{\rho(R_j)^{k_j}} - L_{R_j}\right\|
\end{aligned}$$

The last inequality results from the submultiplicative axiom and the triangle inequality for vector norms. Since c_{k_j} converges to c and $[\rho(R_j)^{-1}\cdot R_j]^{k_j}$ converges to L_{R_j} for $k_j \to \infty$, the right-hand side of the inequality converges to 0. We conclude that

$$\lim_{k_j\to\infty}\alpha_{k_j}(PV_{n-1})\cdot PV_{n-1}\cdot R_j^{k_j} = c\cdot PV_{n-1}\cdot L_{R_j} \tag{5}$$

which completes the proof. □

From Proposition 2 we now have that the present row vector PV_N converges to a particular limit distribution. This limit distribution can in fact be directly computed from equation (5) from the proof of the proposition. Horn and Johnson [11, Lemma 8.2.7] further provide an upper bound on the rate of the convergence to this limit distribution

$$\left\|[\rho(R_j)^{-1}\cdot R_j]^{k_j} - L_{R_j}\right\|_\infty < d\cdot r^{k_j} \tag{6}$$

for some positive constant $d \leq 1$ which depends on R_j and for any r with

$$\frac{|\lambda_2|}{\rho(R_j)} < r < 1 \tag{7}$$

where λ_2 is the second largest modulus eigenvalue of R_j.

From the upper bound on the rate of convergence of the present row vector, we can now establish, for any level of accuracy θ, the value of k_j for which the right-hand side of equation (6) becomes smaller than θ, that is, the value of k_j for which the present row vector will converge to the limit distribution within θ. The importance of this result lies in the observation that for a given Markovian model, we can determine, before actually obtaining any evidence, the number of consecutive similar observations for which the probability distribution of the modelled process will converge within a wished-for level of accuracy. We can then forestall performing inference whenever this number is exceeded. Figure 5 summarises our scheme for inference in Markovian models with consecutive similar observations. Note that the inference is resumed as soon as a dissimilar observation is found after a sequence of similar observations. The inference then is resumed using the approximate row vector which may include an error of most θ.The error in this vector decreases exponentially over time. The rate of the decrease depends on the mixing properties of the transition matrix P of the process; we refer to [4] for further details.

for each value j determine k'_j such that $d \cdot r^{k'_j} < \theta$ where θ is the specified level of accuracy;

for $n = 1, \ldots, N$ **do**

$j \leftarrow$ observed value at time n;

if $n > k'_j$ and the observations at times $n - k'_j - 1, \ldots, n - 1$ equal j **then**

$PV_n = PV_{n-1}$

else

$PV_n = \alpha \cdot PV_{n-1} \cdot P \cdot OM(j)$ where α is a normalisation constant

Fig. 5. Pseudocode for monitoring with consecutive similar observations.

As an example, we consider again our model of pneumonia from Figure 1 and the data for patient Id.1051. For the combination of observations $BT > 38.5$ and SA=yes we find that

$$R_j = \begin{bmatrix} 0.95 & 0.05 \\ 0.15 & 0.85 \end{bmatrix} \cdot \begin{bmatrix} 0.56 & 0 \\ 0 & 0.028 \end{bmatrix} \cdot \begin{bmatrix} 0.708 & 0 \\ 0 & 0.55 \end{bmatrix} = \begin{bmatrix} 0.3767 & 0.0008 \\ 0.0595 & 0.0131 \end{bmatrix}$$

From the computed matrix R_j we thus have that $\rho(R_j) = 0.3768$ and $\lambda_2 = 0.0131$. From equation (7), we now find that the rate of convergence is approximately 0.0345, which means that the speed with which the present row vector approaches its limit distribution is quite high. In fact, for any level of accuracy $\theta \geq 0.001$, the number of consecutive similar observations for which inference has to be performed equals at most $k = 3$. We thus find that, for this patient, the probability distribution of pneumonia does not change by more than θ after time 18:00. Any additional similar observations can therefore be disregarded upon inference.

Similar results hold also for the other two types of inference in Markovian models. For forecasting, it is evident from equation (1) that as long as the present vector PV_N converges, the future row vector $FV_{n,N}, n > N$, converges as well. With regard to smoothing, we observe from equation (2) that the matrix $AP^{n+1,n}$ remains bounded when $p(X_n)$ converges, because the ratio $p(X_n)/p(X_{n+1})$ remains bounded. Since the matrix $AP^{n+1,n}$ stays bounded, the backward row vector $BV_{n,N}, n < N$, will converge.

5 Markovian models with richer structure

The essence of our analysis for HMMs extends to Markovian models in general. These models can have a richer structure either in the observable variables or in the hidden variables, or in both. In this section we briefly review the extension of our analysis to these different types of model.

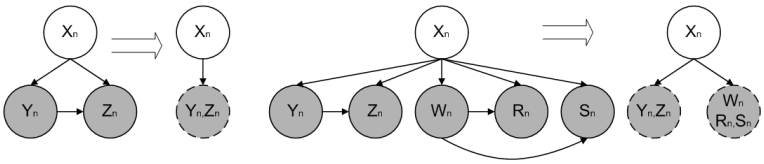


Fig. 6. Markovian models with different structures in their observable variables; the grey nodes represent the observable variables and the dotted nodes represent compound variables.

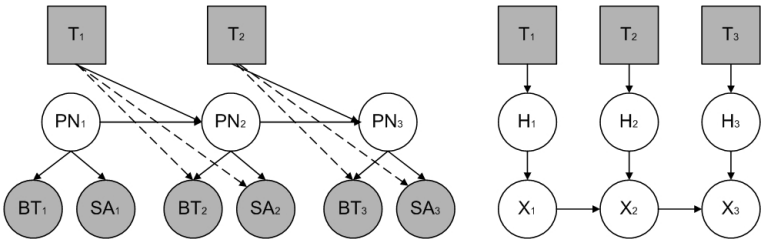


Fig. 7. The effect of input variables T_n on the hidden process.

5.1 Structure in the observable variables

The simplest extension of our analysis pertains to Markovian models with multiple observable variables that are conditionally independent given the hidden variable. Each such observable variable $Y_n^k, k = 1, \ldots, \xi$, has associated observation column matrices $OM_k(j_k)$ for its possible values j_k. Upon inference we now have, for each time n, a set of observations corresponding with the separate observable variables. We then use the product matrix $OM(j_1, \ldots, j_\xi) = \prod_{k=1}^{\xi} OM_k(j_k)$ in the various computations. Our motivating example illustrates a model with such multiple observable variables. Note that Markovian models with multiple observables that are independent given the hidden variable can be considered as dynamic extensions of Naive Bayesian classifiers, where the focus is to distinguish between various classes based on a collection of observations [10]. If the observation variables exhibit some mutual dependencies as in Figure 6, we can construct an observation matrix that describes the joint distribution over these variables. This matrix then is looked upon as the observation matrix of a single compound variable with the joint value assignments of the included variables for its values. Note that the new observation matrix can become very large for multiple observable variables that can take many values.

The dynamics of the hidden variable of a Markovian model may depend on the evolution of another variable. Such models have been called input-output models in the speech recognition literature [3]. Similar models have been used for decision planning in medicine [18], where the input is an action variable modelling alternative treatments. As an example, Figure 7a depicts a Markovian model with an input variable T_n for our example domain of application. In general, a Markovian model with input variables $\mathbf{T}_n$ has associated a *conditional transition matrix*

$P_{X|\mathbf{T}_n}$, which in essence is a set of transition matrices for the evolution of the hidden variable, given each combination of values for the input variables. Whenever the input and observable variables of a model are jointly observed to have the same combination of values, we can use the conditional transition matrix to perform an analysis similar to the one in the previous sections.

To conclude, we consider models in which an input variable affects the hidden process through another hidden variable. Figure 7b illustrates such a model. For these models also, as long as the input variable is observed to consecutively have the same value, the probability distribution of the hidden process converges to a limit distribution within a predefined level of accuracy. The rate of the convergence however depends on the properties of a matrix that consists of a linear combination of the conditional transition matrices that represent the influence of the hidden variable on the hidden process. More specifically, in the model of Figure 7b, the input variable T affects the hidden variable H which subsequently affects the hidden process X through the conditional transition matrix $P_{X|h_i}$ for each value h_i of H. With $s \geq 1$ consecutive input observations $T = t$ starting at time n, the present row vector will converge with a rate proportional to the ratio $|\lambda_2|/\rho(A)$, where λ_2 is the second largest modulus eigenvalue of the matrix $A = \sum_i p(H = h_i \mid T = t) \cdot P_{X|h_i}$ that represents the unconditional transitional behaviour of the hidden process X.

5.2 Structure in the observable and hidden variables

Another extension of our analysis pertains to Markovian models in which separate subnetworks can be distinguished that are conditionally independent given the hidden variable. For ξ conditionally independent subnetworks B_i with the observable variables $\mathbf{Y}_{\mathbf{B_i}}, i = 1 \ldots, \xi$, we then use in the various computations the matrix $OM(\mathbf{Y}_{\mathbf{B_1}}, \ldots, \mathbf{Y}_{\mathbf{B}_\xi}) = \prod_{i=1}^{\xi} OM_{B_i}(\mathbf{Y}_{\mathbf{B_i}})$, where $OM_{B_i}(\mathbf{Y}_{\mathbf{B_i}}) = p(\mathbf{Y}_{\mathbf{B_i}} \mid X_n)$ captures the influence of the observations in the i-th subnetwork on the posterior distribution of the hidden variable.

So far, we assumed that the sequences of consecutive similar observations involved *all* the observable variables. Dependent upon the topological properties of the model, however, our analysis also applies to sequences of similar observations that involve only some of the observable variables. We recall that the concept of *d-separation* [17] provides for reading independencies off the graphical structure of a Markovian model. A subset $\mathbf{H}_n$ of the hidden variables may then be d-separated by a set of observable variables $\mathbf{Y}_n$ from another set of observable variables $\mathbf{Z}_n$; Figure 8 illustrates the basic idea. The set $\mathbf{Z}_n$ upon observation then cannot affect the probability distributions of the hidden variables in $\mathbf{H}_n$. Our analysis now applies directly to similar consecutive observations for the observable variables that are not d-separated from $\mathbf{H}_n$.

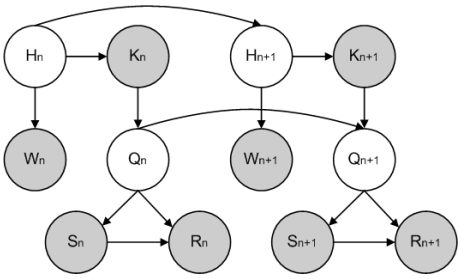


Fig. 8. The hidden variable H_n is independent of the set of observable variables $\mathbf{Z}_n = \{S_n, R_n\}$ as long as $\mathbf{Y}_n = \{K_n\}$ is observed. Our analysis holds for any sequence of similar consecutive observations for W_n, K_n, regardless of the observations for $\mathbf{Z}_n$.

6 Computational savings

In the previous two sections, we have argued that the observation of consecutive similar values for the observable variables in a Markovian model can be exploited to forestall part of the inference. To investigate the ensuing computational savings, we monitor the example Markovian model from Figure 1 as well as the real-life *dVAP* and *Mildew* models.

6.1 The example Markovian model

For the example Markovian model from Figure 1, we briefly address the computational savings that can be achieved upon runtime by exploiting consecutive similar observations. We begin by observing that, if the hidden variable has m possible states, monitoring requires $O(m^2)$ operations per time slice. Smoothing requires $O(m^2 \cdot N)$ operations for a dataset with observations up to and including time N; smoothing further needs $O(m \cdot N)$ space to store the matrices AP that will be used to compute the backward row vector. We now suppose that for our model we have available a dataset that includes q sequences of $s_i,\ i = 1, \ldots, q$, consecutive similar combinations of observations, respectively. We further suppose that out of these q sequences, there are π different combinations, each with its own value $k_j,\ j = 1, \ldots, \pi$, for the number of observations that need to be propagated; each such combination occurs λ_j times, so that $\sum_{j=1}^{\pi} \lambda_j = q$. For the sequence i of the jth combination of observations, therefore, we do not need to perform inference for $s_i - k_j$ time slices. For the dataset under study, we will thus perform inference for $[N - (\sum_{i=1}^{q} s_i - \sum_{j=1}^{\pi} \lambda_j \cdot k_j)]$ time slices with our new scheme, compared to the N slices that would be performed with an exact algorithm.

To study the computational savings in a more practical yet controlled setting, we generated three datasets. Each dataset concerns a period of three weeks and therefore includes $3 \cdot 7 \cdot 12 = 252$ combinations of observations for the two variables *BT* and *SA*. Each dataset further has been generated to contain sequences of similar

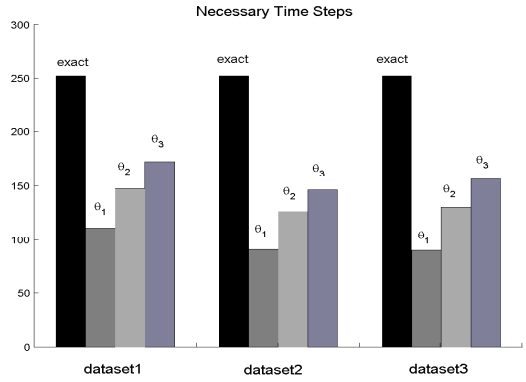


Fig. 9. The number of time slices performed by exact inference and by approximate inference for different levels of accuracy.

	dataset 1	dataset 2	dataset 3
θ_1	55.19%	62.60%	62.97%
θ_2	41.15%	48.92%	47.44%
θ_3	31.43%	41.06%	37.36%

Table 2
The percentage of savings in space requirements compared to exact inference.

observations of lengths $6, 8$, and 10. Dataset 1 has 12 such sequences of length 6, 10 sequences of length 8, and 8 sequences of length 10; for the second dataset, these numbers are $8, 12$ and 10 respectively, and for the third dataset they are $10, 8$ and 12. With each dataset, we performed exact inference using our alternative framework; we further performed approximate inference as described above using the levels of accuracy $\theta_1 = 0.01, \theta_2 = 0.001$ and $\theta_3 = 0.0001$. The experiments were run on a 2.4 GHz Intel(R) Pentium computer, using Matlab 6.1. Figure 9 shows the number of time slices for which the computations are conducted per dataset. We note that the number of time slices for which inference is performed, is reduced for all the datasets by approximately 60% with θ_1, by 45% with θ_2, and by 30% with θ_3. Table 2 shows the savings in space requirements upon runtime per dataset for the different levels of accuracy. We note that increasing the accuracy by one order of magnitude results in a $10-15\%$ increase in space savings. The results thus reveal considerable savings and suggest that longer sequences of observations and a lower wished-for accuracy may lead to larger savings in both time and space requirements. For valid statistical conclusions, however, more experimental results are necessary.

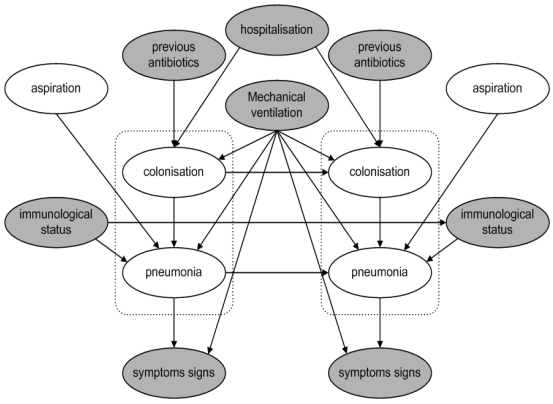


Fig. 10. The dVAP model for the diagnosis of VAP for two consecutive time slices; clear nodes are hidden, shaded nodes are observable. The dashed boxes indicate the hidden processes of the model.

6.2 The dVAP model

The *dVAP* model is a DBN that has recently been constructed for diagnosing VAP in ICU patients [5]. Figure 10 gives a compact representation of the dVAP model, where each time slice represents a single day. The model has been developed with the help of a single infectious disease specialist and has been evaluated for a period of 10 days on a group of 20 patients drawn from the files of the ICU of the University Medical Centre Utrecht in the Netherlands, 5 of whom were diagnosed with VAP.

The dVAP model includes two hidden processes (*colonisation* and *pneumonia*) that interact with each other, three input processes (summarised in *immunological status*), three input observable variables (*hospitalisation*, *mechanical ventilation*, and *previous antibiotics*), and seven output observable variables (summarised in *symptoms-signs*). In each time slice, the model includes a total of 30 variables. Each of the interacting processes consists of seven subprocesses that are a-priori independent. In total, there are 17 variables that belong to the forward interface of the model and there are 17 binary hidden variables per time slice. The runtime complexity of the interface algorithm for exact inference in the dVAP model thus is between $\Omega(2^{18})$ and $O(2^{34})$, showing that inference is quite time consuming if not infeasible.

From the topological structure of the dVAP model we notice that we need to obtain consecutive similar observations for all the observable variables to allow for reducing the computational burden of inference. The application under study further allows consecutive similar observations for the last four days of the observation period only, because *mechanical ventilation* changes periodically until the sixth day after ICU admission, before it remains unaltered. We found that for six out of our collection of 20 patients, there were at least two consecutive days during which the same combination of values for the observable variables was obtained. Table 3

day	m.v	x.t	i/e.r	b.t	a.d	s.a	s.p	l.	p.d	i.	c.	h.	p.a
1	5	1	-	-	2	2	2	-	2	2	2	1	30
2	4	-	2	1	2	1	2	1	2	2	2	1	2
3	3	-	1	2	2	1	2	2	2	2	2	1	2
4	3	-	1	2	2	1	2	1	2	2	2	1	2
5	2	1	2	2	2	1	2	1	2	2	2	1	2
6	2	1	2	2	1	1	1	2	2	2	2	2	2
7	1	1	1	2	1	1	2	2	2	2	2	2	30
8	1	2	1	2	1	1	2	2	2	2	2	2	30
9	1	2	1	2	1	1	2	2	2	2	2	2	30
10	1	2	2	2	1	1	2	2	2	2	2	2	30

Table 3
The dataset for patient Id.25724, where the names of the observable variables have been abbreviated and the numbers stand for their different values respectively; see [5] for more details.

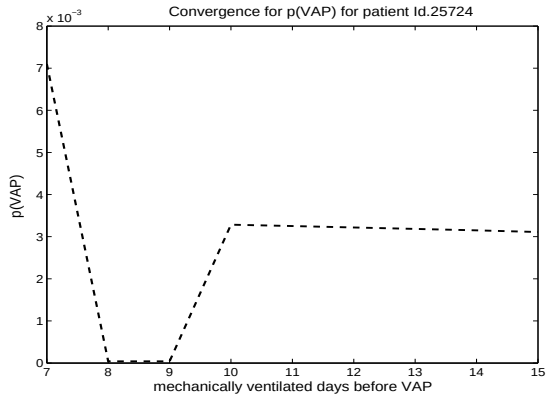


Fig. 11. The convergence behaviour in the probability of pneumonia for patient Id.25724 for two periods of consecutive similar combinations of observations.

presents the dataset for one of these six patients.

Upon studying the probability distributions of the two hidden processes for the six patients, we found that on average after three days of consecutive similar observations these distributions had converged to a limit distribution for any level of accuracy $\theta \geq 0.0028$. More specifically, each of the subprocesses of the two hidden processes had converged to a limit distribution and could thus be disregarded for further inference as long as consecutive similar observations were obtained. The size of the forward interface thereby is reduced from 17 to three and the number of hidden variables reduced from 17 to two. The model-specific runtime complexity of the interface algorithm now lied between $\Omega(2^4)$ and $O(2^5)$. We thus achieved saving of at least an order of 2^{13}. Figure 11 illustrates the convergence behaviour of the probability of pneumonia for a specific patient for whom two different sequences of consecutive similar combinations of observations were obtained; one sequence within the period of days 8-9 and another sequence extending for five days after day 10.

On a 2.4 GHz Intel(R) Pentium computer, exact inference with the dVAP model

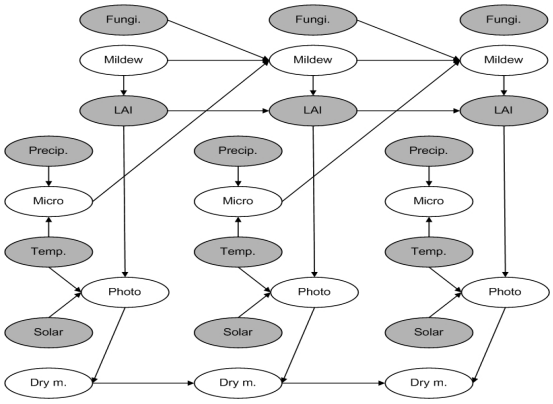


Fig. 12. The Mildew model for forecasting the extension of mildew fungus and the gross yield for three consecutive time slices; clear nodes are hidden, shaded nodes are observable.

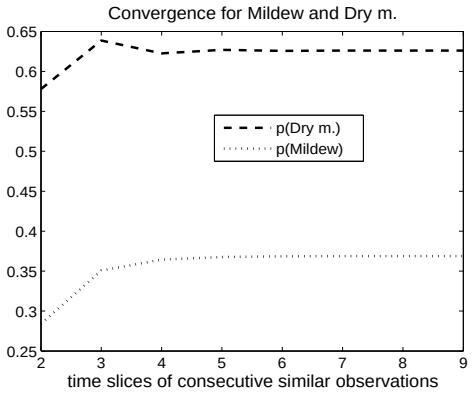


Fig. 13. The convergence behaviour in the probability of Mildew and Dry m. for consecutive similar observations.

took about two and a half minutes for 10 time slices, with an average of 0.25 minutes per slice. For a patient who is observed for one month and for whom two sequences of 10 days of consecutive similar combinations of observations are found, the inference time is reduced from approximately 7.5 minutes to 4 minutes, which is a 47% reduction of the runtime requirements. We feel that especially for datasets of patients who have a long observation period and for whom several sequences of consecutive similar combinations of observations are found, the runtime savings can be substantial.

6.3 The Mildew model

The *Mildew* model is a DBN for forecasting the extension of mildew fungus and the gross yield from a wheat field [13]. The Mildew model has nine variables per time slice, four of which are hidden (*Mildew, Micro climate, Photo-synthesis* and *Dry matter*), four are input observable variables (*Fungicide, Precipitation, Temperature* and *Solar energy*), and one is an output observable variable (*Leaf Area Index*). Figure 12 depicts the model, where the names of the variables have been abbreviated

for readability. Our focus is on determining the probability distributions of the variables *Mildew* and *Dry matter* over time. We notice that *Dry matter* is d-separated from *Mildew* given the *Leaf Area Index*. For computing the probability distribution of *Dry matter*, therefore, we need to take into account only the observations for the variables *Leaf Area Index, Solar energy* and *Temperature*. When the values for these variables are consecutively observed to stay the same, the probability distribution for *Dry matter* will converge no matter what the values for the other two observable variables, *Fungicide* and *Precipitation*, are. The rate of the convergence can be determined using the conditional transition matrix $P_{\text{Dry m.}|\text{Photo}}$. Similarly the values for the observable variable *Solar energy* do not influence the probability distribution of *Mildew* as long as similar values are observed for the variables *Fungicide, Leaf Area Index, Precipitation* and *Temperature*.

For binary variables, the runtime complexity of inference in the Mildew model using the interface algorithm is between $\Omega(2^6)$ and $O(2^9)$. When the values for all the observable variables are the same for consecutive time slices, the complexity of the inference is reduced to between $\Omega(2^2)$ and $O(2^3)$ after convergence has been established. We generated random parameters and observations for the Mildew model to study the convergence behaviour of the probability distributions of the hidden processes of *Mildew* and *Dry matter* when similar values are obtained consecutively for the variables that influence these probability distributions respectively. Figure 13 illustrates the behaviour that we found. We notice that after obtaining four consecutive similar observations, the probability distributions of both variables have converged within a level of accuracy $\theta = 0.0043$; for the variable *Mildew* in fact, a level of accuracy as small as $\theta = 0.0033$ is guaranteed.

7 Conclusions

Inference in Markovian models such as DBNs and HMMs is hard in general. Algorithms for exact inference in fact are practically infeasible for many real-life applications due to their high computational complexity. We have shown, that the nature of the observations obtained can sometimes be exploited to reduce the computational requirements of inference upon runtime. We have studied more specifically the effect of consecutive similar observations on the posterior distribution of a hidden process, and have shown theoretically that it will converge to a limit distribution within some level of accuracy. Observing further similar values will therefore not alter the distribution beyond this level and no further inference is required. We have presented an algorithm that builds upon these results and forestalls inference as soon as possible.

We have introduced a realistic example from the medical domain that motivated our analysis. Experimental evaluation of our ideas on the example has shown promising results with respect to the computational savings that can be achieved upon

runtime. We have further demonstrated how our analysis can be extended to Markovian models with richer structure in the observable or hidden space and discussed the potential of reducing the runtime requirements for inference in such models. To validate our theoretical results, we further experimented with two larger real-life Markovian models. We showed that upon obtaining consecutive similar observations for the dVAP model, the runtime requirements of inference can be reduced considerably allowing for runtime savings in the computations involved. For the Mildew model, we showed how our results can be exploited to reduce the computational requirements upon runtime when only a subset of the observable variables is consecutively observed to have the same value. We conclude that for monitoring-like applications where it is not unlikely that consecutive similar observations are obtained, our results provide for speeding up the inference procedure by forestalling some of the computations involved.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments on an earlier version of this manuscript.

References

[1] S. Andreassen, R. Hovorka, J. Benn, K.G. Olesen, and E.R. Carson. A model-based approach to insulin adjustment. *Proceedings of the 3rd Conference on Artificial Intelligence in Medicine*, pp. 239-248, 1991.

[2] L. E. Baum, T. Peterie, G. Souled, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics* 41(1): 164-171, 1970.

[3] Y. Bengio and P. Frasconi. Input/Output HMMs for sequence processing. *IEEE Transactions on Neural Networks* 7(5): 1231-1249, 1996.

[4] X. Boyen and D. Koller. Tractable inference for complex stochastic processes. *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, pp. 33-42, 1998.

[5] T. Charitos, L.C. van der Gaag, S. Visscher, K. Schurink, and P. Lucas. A dynamic Bayesian network for diagnosing ventilator-associated pneumonia in ICU patients. *Proceedings of the 10th Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, pp. 32-37, 2005.

[6] Y. Chauvin and P. Baldi. Hidden markov models of the g-protein-coupled receptor family. *Journal of Computational Biology* 1(4): 311-336, 1994.

[7] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley, 1991.

[8] R.G. Cowell, A.P. Dawid, S.L. Lauritzen, and D.J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer, 1999.

[9] T. Dean and K. Kanazawa. A model for reasoning about persistence and causation. *Computational Intelligence* 5: 142-150, 1989.

[10] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29: 131-163, 1997.

[11] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge: University Press, 1990.

[12] D.L. Isaacson and R.W. Madsen. *Markov Chains: Theory and Applications*. Wiley and Sons, 1976.

[13] U. Kjaerulff. dHugin: A computational system for dynamic time-sliced Bayesian networks. *International Journal of Forecasting*, 11: 89-111, 1995.

[14] P.J. Lucas, N.C. de Bruijn, C. Schurink, and A. Hoepelman. A probabilistic and decision theoretic approach to the management of infectious disease at the ICU. *Artificial Intelligence in Medicine* 19(3): 251-279, 2000.

[15] I.L. MacDonald and W. Zucchini. *Hidden Markov and Other Models for Discrete-valued Time series*. Chapman and Hall, 1997.

[16] K.P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. Ph.D. diss, University of California Berkeley, 2002.

[17] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Palo Alto, 1988.

[18] N.B. Peek. Explicit temporal models for decision-theoretic planning of clinical management. *Artificial Intelligence in Medicine* 15(2): 135-154, 1999.

[19] L.R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE* 77(2): 257-286, 1989.

[20] S. Ross. *Stochastic Processes*. Wiley and Sons, second edition, 1996.

[21] C.A.M. Schurink. *Ventilator Associated Pneumonia: a Diagnostic Challenge*. Ph.D. diss, Utrecht University, 2003.

[22] R. Shachter. Probabilistic inference and influence diagrams. *Operations Research* 36(4): 589-604, 1988.

[23] P. Weber and L. Jouffe. Reliability modelling with dynamic Bayesian networks. *Proceedings of the 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pp. 57-62, 2003.