

Article

A Simplified Matrix Formulation for Sensitivity Analysis of Hidden Markov Models

Seifemichael B. Amsalu ¹, Abdollah Homaifar ^{1,*} and Albert C. Esterline ²

¹ Department of Electrical and Computer Engineering, North Carolina A&T State University, Greensboro, NC 27411, USA; samsalu@aggies.ncat.edu

² Department of Computer Science, North Carolina A&T State University, Greensboro, NC 27411, USA; esterlin@ncat.edu

* Correspondence: homaifar@ncat.edu; Tel.: +1-336-285-3709

Received: 9 June 2017; Accepted: 14 August 2017; Published: 22 August 2017

Abstract: In this paper, a new algorithm for sensitivity analysis of discrete hidden Markov models (HMMs) is proposed. Sensitivity analysis is a general technique for investigating the robustness of the output of a system model. Sensitivity analysis of probabilistic networks has recently been studied extensively. This has resulted in the development of mathematical relations between a parameter and an output probability of interest and also methods for establishing the effects of parameter variations on decisions. Sensitivity analysis in HMMs has usually been performed by taking small perturbations in parameter values and re-computing the output probability of interest. As recent studies show, the sensitivity analysis of an HMM can be performed using a functional relationship that describes how an output probability varies as the network's parameters of interest change. To derive this sensitivity function, existing Bayesian network algorithms have been employed for HMMs. These algorithms are computationally inefficient as the length of the observation sequence and the number of parameters increases. In this study, a simplified efficient matrix-based algorithm for computing the coefficients of the sensitivity function for all hidden states and all time steps is proposed and an example is presented.

Keywords: sensitivity analysis; hidden Markov models; Bayesian network

1. Introduction

A Hidden Markov Model (HMM) is a stochastic model of the dynamic process of two related random processes that evolve over time. A set of stochastic processes that produces the sequence of observed symbols is used to infer an underlying stochastic process that is not observable (hidden states). HMMs have been widely utilized in many application areas including speech recognition [1], bioinformatics [2], finance [3], computer vision [4], and driver behavior modeling [5,6]. A comprehensive survey on the applications of HMMs is presented in [7]. A simple dynamic Bayesian network can be used to represent a stationary HMM with discrete variables [8,9]. Consequently, algorithms and theoretical results developed for dynamic Bayesian networks can be applied to HMMs as well.

Sensitivity analysis in HMMs has been done by perturbation analysis in which a brute-force computation of the effect of small changes on the output probability of interest is done [10,11]. Sensitivity analysis research has shown that a functional relationship can be established between a parameter variation and the output probability of interest. This function can represent the effect of any change in the parameter under consideration as compared to the perturbation analysis. Consequently, the goal of sensitivity analysis becomes developing techniques to create this function. This general function will also enable us to compute bounds on the output probabilities without sensitivity analysis [12–15].

The objective of this work is to develop an algorithm for sensitivity analysis in HMMs directly from the model's representation. Our focus is on parameter sensitivity analysis where the variation of the output is studied as the model parameters vary. In this case, the parameters are the (conditional) probabilities of the HMM model (initial, transition, and observation parameters) and the output is some posterior probability of interest (filtering, smoothing, predicted future observations, and most probable path).

Parametric sensitivity analysis can be used for multiple purposes. The sensitivity analysis can be used to identify those parameters which have significant impact on the output probability of interest. Consequently, the result can be used as a basis for parameter tuning and studying the robustness of the model output to changes in the parameters [16–18].

Many researchers study the problem of sensitivity analysis in Bayesian Networks and HMMs. There are two approaches used to compute the constants of the sensitivity function in the standard Bayesian network inference algorithms. The first method is solving systems of linear equations [19] by evaluating the sensitivity function for different values of the varying parameter θ . The second is based on a differential approach [20] by which the coefficients of a multivariate sensitivity function can be computed from partial derivatives. The tailored version of the junction tree algorithm has been used to establish the sensitivity functions in Bayesian networks more efficiently [21,22]. In [23], sensitivity analysis of Bayesian networks for multiple parameter changes is presented. In [24], credal sets that are known to represent the probabilities of interest are used for sensitivity analysis of Markov chains. In [10,11], sensitivity analysis of HMMs based on small perturbations in the parameter values is presented. A tight perturbation bound is derived and it is shown that the distribution of the HMM tends to be more sensitive to perturbations in the emission probabilities than to the transition and initial distributions. A tailored approach for HMM sensitivity analysis, the Coefficient-Matrix-Fill procedure, is presented in [25,26]. It directly utilizes the recursive probability expressions in an HMM. Consequently, it improves the computational complexity of applying the existing approaches for sensitivity analysis in Bayesian networks to HMMs. In [27], imprecise HMMs are presented as a tool for performing sensitivity analysis of HMMs.

In this paper, we propose a sensitivity analysis algorithm for HMMs using a simplified matrix formulation directly from the model representation based on a recently proposed technique called the Coefficient-Matrix-Fill procedure [26]. Until recently, sensitivity analysis in HMMs has been performed using Bayesian network sensitivity analysis techniques. The HMM is represented as a dynamic Bayesian network unrolled for a fixed number of time slices, and the Bayesian sensitivity algorithms are used. However, these algorithms do not utilize the HMMs' recursive probability formulations. In this work, a simple algorithm based on a simplified matrix formulation is proposed. In this algorithm, to calculate the coefficients of the sensitivity functions, a series of Forward matrices $F^k, k = 1, \dots, t$ are used, where k represents the time slice in the observation sequence. The matrices (Initial, Transition, and Observation) where the corresponding model parameter θ varies are decomposed into the parts independent of and dependent on θ for mathematical convenience. This enables us to compute the coefficients of the sensitivity function at each iteration in the recursive probability expression and implement the algorithm in a computer program. These matrices are computed based on a proportional co-variation assumption. The Observation Matrix O is represented as $n \times n$ diagonal matrices O^t whose v th diagonal entry is $P(y_e^t | x_v^t)$ for each state v and whose other entries are 0 at the time t of the observation sequence. The proposed algorithm is computationally efficient as the length of the observation sequence increases. It computes the coefficients of a polynomial sensitivity function by filling coefficient matrices for all hidden states and all time steps.

The paper is organized as follows. In Section 2, the background on HMM and Sensitivity Analysis in HMM is presented. The details of the proposed algorithm for the filtering probability sensitivity function are explained in Section 3. In Section 4, the sensitivity analysis of the smoothing probability using the proposed algorithm is discussed. Finally, the paper is concluded with a summary of the results achieved and recommendations for future research works.

2. Background

In this section, we present the background on HMMs and Sensitivity Analysis in HMMs. First, we will discuss HMMs, HMM inference tasks and their corresponding recursive probability expressions. Then sensitivity analysis in HMMs is explained based on the assumption of proportional co-variation, and a univariate polynomial sensitivity function whose coefficients the proposed algorithm computes is defined. Here, the variables are denoted by capital letters and their values by lower case letters.

2.1. Hidden Markov Models

An HMM is a stochastic statistical model of a discrete Markov chain of a finite number of hidden variables X that can be observed by a sequence of a set of output variables Y from a sensor or other sources. The probability of transitioning from one state to another in this Markov chain is time-invariant, which makes the model stationary. The observed variables Y are stochastic with the observation (emission) probabilities, which are also time invariant. The overall HMM consists of n distinct hidden states of the Markov chain and m corresponding observable symbols. In general, the observations can be discrete or continuous, however, in this work, we focus on the discrete observations. The variable X has $n \geq 2$ hidden states, denoted by $x_i, 1 \leq i \leq n$, and the variable Y has $m \geq 2$ observable symbols, denoted by $y_j, 1 \leq j \leq m$. Formally, an HMM Λ is specified by a set of parameters (A, O, Γ) that are defined as follows:

1. The prior probability distribution (initial vector) Γ has entries $\gamma_i = p(x_i), 1 \leq i \leq n$, which are the probabilities of state x_i being the first state in the Markov chain.
2. The transition matrix A has entries $a_{i,j} = p(x_j|x_i), 1 \leq i, j \leq n$, which are the probabilities of transit from state x_i to state x_j in the Markov chain.
3. The observation matrix O has entries $o_{i,j} = p(y_j|x_i), 1 \leq i \leq n, 1 \leq j \leq m$, which are the probabilities to observe y_j if the current state is x_i .

An example of an HMM where X has two states and Y three symbols is shown in Figure 1a. The two states are x_1 and x_2 and based on them three symbols y_1, y_2 or y_3 are observed. The parameters of the HMM (including initial vector, transition matrix and observation matrix) are also shown in Figure 1a.

The dynamic Bayesian network representation of the HMM in Figure 1a is shown in Figure 1b, unrolled for T time slices [8,9]. The initial vector, transition matrix and observation matrix of the HMM are represented by the labels Γ, A and O respectively, attached to the nodes in the Bayesian network graph. The superscript for the variables X and Y and their values indicate the time slice under consideration.

In this paper, \mathbf{y}_e^t denotes the actual evidence for the variable Y in time slice t and $\mathbf{y}_e^{t_i:t_j}$ represents a sequence of observations $\mathbf{y}_e^{t_i}, \dots, \mathbf{y}_e^{t_j}$ where \mathbf{e} is a sequence of observations from the set $y_j, 1 \leq j \leq m$. The representation x_i^t denotes the actual state $x_i, 1 \leq i \leq n$, for the variable X in time slice t .

2.1.1. Inference in HMMs

In temporal models, inference means computing the conditional probability distribution of X at time t , given the evidence up to and including time T , that is $p(X^t | \mathbf{y}_e^{1:T})$. The main inference tasks include *filtering* for $T = t$, *prediction* of a future state for $t > T$ and *smoothing*, that is inferring the past for $t < T$. In an HMM, the *Forward-Backward algorithms* are used for the inference tasks and training of the model using an iterative procedure called the *Baum-Welch method (Expectation Maximization)* [28–30]. The *Forward-Backward algorithms* computes the following probabilities for all hidden states i at time $t \leq T$:

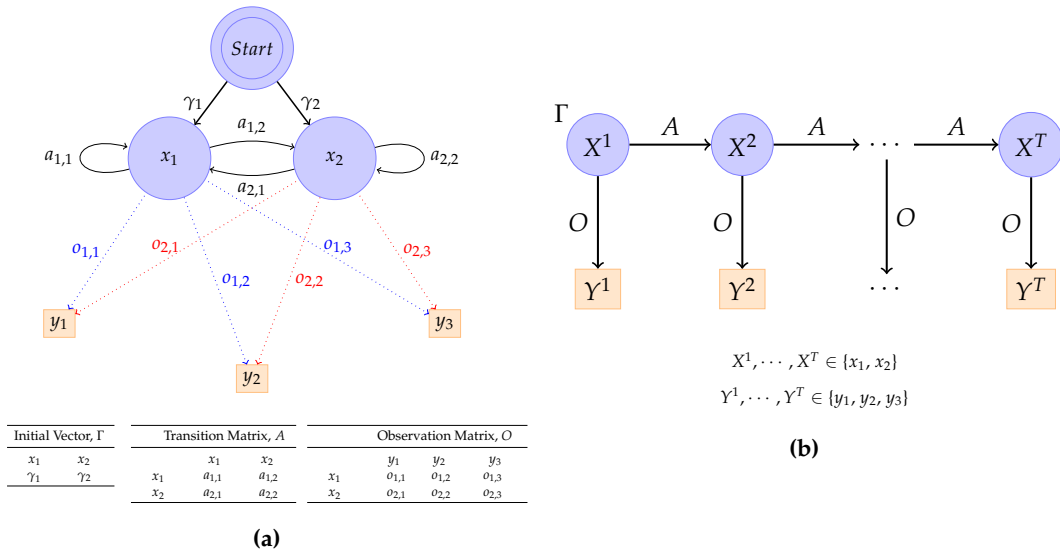


Figure 1. (a) An example of an HMM representation. (b) Its dynamic Bayesian network representation, unrolled for T time slices.

- forward probability $F(i, t) = p(x_i^t, \mathbf{y}_e^{1:t})$, and
- backward probability $B(i, t) = p(\mathbf{y}_e^{t+1:T} | x_i^t)$

resulting in

$$\begin{aligned}
 p(x_i^t | \mathbf{y}_e^{1:T}) &= \frac{p(x_i^t, \mathbf{y}_e^{1:T})}{p(\mathbf{y}_e^{1:T})} = \frac{p(x_i^t, \mathbf{y}_e^{1:T})}{\sum_{j=1}^n p(x_j^t, \mathbf{y}_e^{1:T})} = \frac{p(x_i^t, \mathbf{y}_e^{1:t}) \cdot p(\mathbf{y}_e^{t+1:T} | x_i^t)}{\sum_{j=1}^n p(x_j^t, \mathbf{y}_e^{1:t}) \cdot p(\mathbf{y}_e^{t+1:T} | x_j^t)} \\
 &= \frac{F(i, t) \cdot B(i, t)}{\sum_{j=1}^n F(j, t) \cdot B(j, t)}
 \end{aligned}
 \tag{1}$$

For $T < t$, the algorithm can be applied by taking $B(i, t) = 1$ and adopting the convention that the configuration of an empty set of observations is TRUE, i.e., $\mathbf{y}_e^{T+1:t} \equiv TRUE$, giving

$$F(i, t) = p(x_i^t, \mathbf{y}_e^{1:t}) = p(x_i^t, \mathbf{y}_e^{1:T}, TRUE) = p(x_i^t, \mathbf{y}_e^{1:T})$$

As shown in Equation (1), any conditional probability $p(x_i^t | \mathbf{y}_e^{1:T})$ can be calculated from the joint probabilities $p(x_i^t, \mathbf{y}_e^{1:T})$ for all hidden states i using Bayes' theorem. Since our objective is sensitivity analysis of the HMM output probability of interest for parameter variations, in the remainder of this paper our focus will be on the joint probabilities $p(x_i^t, \mathbf{y}_e^{1:T})$ for all hidden states i .

The other important inference tasks are the *prediction of future observations*, i.e., $p(\mathbf{y}_e^t | \mathbf{y}_e^{1:T})$ for $t > T$, finding the most probable explanation, i.e., $\arg \max_{x_i^{1:t}} p(x_i^{1:t} | \mathbf{y}_e^{1:t})$ using the *Viterbi Algorithm* and learning the HMM parameters as new observations come in using the *Baum-Welch algorithm*, which can be achieved by the *Forward-Backward algorithms*. The *prediction of future observations* can be computed as the fraction of the two probabilities $p(\mathbf{y}_e^t, \mathbf{y}_e^{1:T})$, $t > T$, and $p(\mathbf{y}_e^{1:T})$, which are computed using forward probabilities.

$$p(\mathbf{y}_e^{1:t}) = \sum_{i=1}^n p(x_i^t, \mathbf{y}_e^{1:t}) = \sum_{i=1}^n F(i, t)$$

Note that if $t > T + 1$, then $p(\mathbf{y}_e^t, \mathbf{y}_e^{1:T})$ can be computed by setting all in-between observations \mathbf{y}_e^k , $t > k > T$, to TRUE as above.

2.1.2. Recursive Probability Expressions

In order to establish the sensitivity functions for HMMs directly from the model’s representation, the recursive probability expression for the Forward-Backward algorithm should be investigated. The repetitive character of the model parameters in the Forward-Backward algorithm is used to drive the sensitivity functions. Consequently, in this section the recursive expressions of the Forward-Backward algorithm [29,31] are reviewed briefly.

Filtering. The *filter probability* is $p(x_v^t, \mathbf{y}_e^t)$ for a specific state v of X . This probability is the same as the forward probability $F(v, t)$ in the Forward-Backward algorithm. For time slice $t = 1$, we simply have that

$$p(x_v^1, \mathbf{y}_e^1) = p(\mathbf{y}_e^1 | x_v^1) \cdot p(x_v^1) = o_{v, \mathbf{e}^1} \cdot \gamma_v,$$

where \mathbf{e}^1 corresponds to the symbol of Y that is observed at time 1. For time slices $t > 1$, we exploit the fact that Y^t is independent of Y^1, \dots, Y^{t-1} , given X^t , written $Y^t \perp Y^{1:t-1} | X^t$. Then,

$$p(x_v^t, \mathbf{y}_e^{1:t}) = p(x_v^t, \mathbf{y}_e^{1:t-1}, \mathbf{y}_e^t) = p(\mathbf{y}_e^t | x_v^t) \cdot p(x_v^t, \mathbf{y}_e^{1:t-1})$$

The first factor in the above product corresponds to o_{v, \mathbf{e}^t} ; conditioning the second factor on the n states of X^{t-1} , we find with $X^t \perp Y^{1:t-1} | X^{t-1}$ that

$$p(x_v^t, \mathbf{y}_e^{1:t-1}) = \sum_{z=1}^n p(x_v^t | x_z^{t-1}) \cdot p(x_z^{t-1}, \mathbf{y}_e^{1:t-1}) = \sum_{z=1}^n a_{z,v} \cdot p(x_z^{t-1}, \mathbf{y}_e^{1:t-1})$$

Taken together, we find for $F(v, t) = p(x_v^t, \mathbf{y}_e^{1:t})$ the recursive expression

$$F(v, t) = \begin{cases} o_{v, \mathbf{e}^1} \cdot \gamma_v & \text{if } t = 1 \\ o_{v, \mathbf{e}^t} \cdot \sum_{z=1}^n a_{z,v} \cdot F(z, t - 1) & \text{if } t > 1 \end{cases} \quad (2)$$

Prediction. The *prediction probability* $p(x_v^t, \mathbf{y}_e^{1:T})$ for $t > T$ can be handled by the the Forward-Backward algorithm by computing $F(v, t)$ with an empty set of evidence for $Y^{T+1:t}$. It can be implemented by replacing the term o_{v, \mathbf{e}^t} by 1 for $t > T$ in Equation (2). As a result, for $T = 0$, the *prediction probability* $p(x_v^t, \mathbf{y}_e^{1:T})$ becomes the prior probability $p(x_v^t)$. The prediction task can be seen as a special case of the filtering task.

Smoothing. Finally, we consider a *smoothing probability* $p(x_v^t, \mathbf{y}_e^{1:T})$ with $t < T$. Using $Y^{t+1:T} \perp Y^{1:t} | X^t$ we find that

$$p(x_v^t, \mathbf{y}_e^{1:T}) = p(x_v^t, \mathbf{y}_e^{1:t}, \mathbf{y}_e^{t+1:T}) = p(\mathbf{y}_e^{t+1:T} | x_v^t) \cdot p(x_v^t, \mathbf{y}_e^{1:t})$$

The second term in this product is a filter probability; the first term is the same as the backward probability $B(v, t)$ in the Forward-Backward algorithm. By conditioning the first term on X^{t+1} and exploiting the independences $X^t \perp Y^{t+1:T} | X^{t+1}$ and $Y^{t+1} \perp Y^{t+2:T} | X^{t+1}$ for $T > t + 1$, we find that

$$\begin{aligned} p(\mathbf{y}_e^{t+1:T} | x_v^t) &= \sum_{z=1}^n p(\mathbf{y}_e^{t+1} | x_z^{t+1}) \cdot p(\mathbf{y}_e^{t+2:T} | x_z^{t+1}) \cdot p(x_z^{t+1} | x_v^t) \\ &= \sum_{z=1}^n o_{z, \mathbf{e}^{t+1}} \cdot a_{v,z} \cdot p(\mathbf{y}_e^{t+2:T} | x_z^{t+1}) \end{aligned}$$

For $t + 1 = T$, this results in

$$p(\mathbf{y}_e^{T:T} | x_v^{T-1}) = \sum_{z=1}^n p(\mathbf{y}_e^T | x_z^T) \cdot p(x_z^T | x_v^{T-1}) = \sum_{z=1}^n o_{z, \mathbf{e}^T} \cdot a_{v,z}$$

Taken together, we find for $B(v, t) = p(\mathbf{y}_e^{t+1:T} | x_v^t)$ the recursive expression

$$B(v, t) = \begin{cases} \sum_{z=1}^n o_{z,e^t} \cdot a_{v,z} & \text{if } t = 1 \\ \sum_{z=1}^n o_{z,e^{t+1}} \cdot a_{v,z} \cdot B(z, t + 1) & \text{if } t > 1 \end{cases} \quad (3)$$

In this work, we present the sensitivity of the HMM for filtering and smoothing probabilities for transition, initial and observation parameter variation. Therefore, the above discussion on the recursive probability expression for the filtering probability, Equation (2), and smoothing probability Equation (3), will be used to formulate our algorithm.

2.2. Sensitivity Analysis in HMM

As shown in recent studies [13,25,32,33], sensitivity analysis is establishing a functional relationship that describes how output probability varies as the network’s parameters change. A posterior marginal probabilities, $y = p(v|e)$, where v is value of a variable V and e is the evidence available, is considered to represent the output probabilities of interest. The network parameter under consideration is represented as $\theta = p(v_j|\pi)$ for a value v_j of a variable V , and π is an arbitrary combination of the values of the set of evidences of V . So $p(v|e)(\theta)$ represents the posterior marginal probabilities as a function of θ .

Consequently, when the parameter $\theta = p(v_j|\pi)$ varies, each of the other conditional probabilities $\theta = p(v_i|\pi)$ from the same distribution co-vary accordingly by the ratio between the the remaining probabilities. That is, if a parameter $\theta = p(v_j|\pi)$ for a variable V is varied, then

$$p(v_i|\pi)(\theta) = \begin{cases} \theta & \text{if } i = j \\ p(v_i|\pi) \cdot \frac{1-\theta}{1-p(v_j|\pi)} & \text{if } i \neq j \end{cases} \quad (4)$$

The co-variation simplifies to $p(v_i|\pi)(\theta) = 1 - \theta$ for binary-valued V . The sensitivity function is defined based on the stated proportional co-variation assumption. The constants in the general form of the sensitivity function are generated from the network’s parameters that are neither varied nor co-varied, and their values depend on the output probability of interest. A sensitivity function for a posterior probability of interest is a quotient of two polynomial functions, since $p(v|e) = p(v e) / p(e)$, and hence a rational function.

We have the following univariate polynomial sensitivity function to define the joint probability of a hidden state and a sequence of observations as a function of a model parameter θ .

$$p(x_v^t, \mathbf{y}_e^{1:T}) = \sum_{i=0}^N c_{v,i}^t \cdot \theta^i \quad (5)$$

The coefficients $c_{v,i}^t$ are derived from the network parameters that are not co-varied by assuming proportional co-variation of the parameters from the same conditional distribution. The coefficients $c_{v,i}^t$ depend on the hidden state v and time slice t under consideration (see for details [25,26]).

3. Sensitivity Analysis of Filtering Probability

In this section, the simplified matrix formulation for sensitivity of the filtering probability to the variation of the initial, transition and observation parameters is proposed. Let us consider the sensitivity functions $p(x_v^t, \mathbf{y}_e^{1:t})(\theta)$ for a filtering probability and the variation of one of the model parameters, viz., θ . From Equation (2), the forward recursive filtering probability expression, it follows that

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta) = \begin{cases} o_{v,e^1}(\theta) \cdot \gamma_v(\theta) & \text{if } t = 1 \\ o_{v,e^t}(\theta) \cdot \sum_{z=1}^n a_{z,v}(\theta) \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta) & \text{if } t > 1 \end{cases} \quad (6)$$

where

$$\gamma_v(\theta) = \begin{cases} \gamma_v(\theta_\gamma) & \text{for initial parameter variation } \theta = \theta_\gamma \\ \gamma_v & \text{otherwise} \end{cases}$$

$$a_{z,v}(\theta) = \begin{cases} a_{z,v}(\theta_a) & \text{for transition parameter variation } \theta = \theta_a \\ a_{z,v} & \text{otherwise} \end{cases}$$

and

$$o_{v,e^t}(\theta) = \begin{cases} o_{v,e^t}(\theta_o) & \text{for observation parameter variation } \theta = \theta_o \\ o_{v,e^t} & \text{otherwise} \end{cases}$$

The sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta)$ coefficients are computed by the proposed method using a series of Forward matrices $F^k, k = 1, \dots, t$. The sensitivity of the filtering probability for transition, initial and observation parameter variations is formulated in the next subsections.

3.1. Transition Parameter Variation

In this subsection, the simplified matrix formulation (SMF) for sensitivity of the filtering probability to variation of the transition parameter is proposed. Let us consider the sensitivity functions $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_a)$ for a filtering probability and transition parameter $\theta_a = a_{r,s}$. From Equation (6), the recursive filtering probability sensitivity function, it follows that for $t = 1$ we have the constant $p(x_v^1, \mathbf{y}_e^1)(\theta_a) = o_{v,e^1} \cdot \gamma_v$, and for $t > 1$,

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_a) = o_{v,e^t} \cdot \sum_{z=1}^n a_{z,v}(\theta_a) \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_a) \tag{7}$$

In the next sub-subsections, the formulation of the proposed method to compute the coefficients of the sensitivity function in Equation (5) is presented in detail. First, the decomposition of the transition matrix into the parts of A independent of and dependent on θ_a and the observation matrix representation for the simplified matrix formulation is presented. Then the SMF representation for the sensitivity of the filtering probability to transition parameter variation and its algorithm implementation are discussed.

3.1.1. Transition Matrix Decomposition and Observation Matrix Representation

In the proposed algorithm, to calculate the coefficients in a series of Forward matrices $F^k, k = 1, \dots, t$, the Transition Matrix A is divided into \bar{A} and \hat{A} that are independent of and dependent on θ_a , respectively, for mathematical convenience. The matrices \bar{A} and \hat{A} are computed based on the proportional co-variation assumption shown in Equation (4). Let us explain the decomposition considering an HMM with an $n \times n$ Transition Matrix A with the transition parameter $\theta_a = a_{r,s} = a_{n,1} = p(x_1^t | x_n^{t-1})$. This makes the transition matrix A a function of θ_a , and, with the proportional co-variation assumption shown in Equation (4), it becomes

$$A(\theta_a) = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_a & a_{n,2} \cdot \frac{1-\theta_a}{1-a_{n,1}} & \cdots & a_{n,n} \cdot \frac{1-\theta_a}{1-a_{n,1}} \end{bmatrix}$$

This matrix is divided into \bar{A} and \hat{A} that are independent of and dependent on θ_a , respectively, as follows, and such that $A(\theta_a) = \bar{A} + \hat{A}(\theta_a)$.

$$\bar{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \frac{a_{n,2}}{1-a_{n,1}} & \cdots & \frac{a_{n,n}}{1-a_{n,1}} \end{bmatrix} \quad \text{and} \quad \hat{A}(\theta_a) = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{-a_{n,2}}{1-a_{n,1}} & \cdots & \frac{-a_{n,n}}{1-a_{n,1}} \end{bmatrix} \cdot \theta_a$$

In short, this decomposition can be represented in the algorithm simply as \bar{A} and \hat{A} that are independent of and dependent on θ_a , respectively.

In the simplified matrix representation, the Observation Matrix O is represented as an $n \times n$ diagonal matrices O^t whose v^{th} diagonal entry is $P(y_e^t|x_v^t)$ for each state v and whose other entries are 0 at the time t of the observation sequence [31]. Let us explain this representation by considering the HMM with an $n \times m$ Observation Matrix O . Given the following sequence of observations: y_2^1, y_1^2 and y_3^3 , the corresponding diagonal matrices become

$$O^1 = \begin{bmatrix} o_{1,2} & 0 & \cdots & 0 \\ 0 & o_{2,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & o_{n,2} \end{bmatrix} = \text{diag}(O_{:,2}), \quad O^2 = \text{diag}(O_{:,1}) \quad \text{and} \quad O^3 = \text{diag}(O_{:,3})$$

3.1.2. SMF for Sensitivity Analysis of the Filtering Probability to the Transition Parameter Variation

In the Simplified Matrix Representation (SMF), the sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_a)$ shown in Equation (7) can be written as a simple matrix-vector multiplication. It follows that for $t = 1$ we have the constant $p(x_v^1, \mathbf{y}_e^1)(\theta_a) = o_{v,e^1} \cdot \gamma_v$, which becomes $p(X^1, \mathbf{y}_e^1)(\theta_a) = O^1 \Gamma^\top$ in the SMF. For $t > 1$, we have

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_a) = o_{v,e^t} \cdot \sum_{z=1}^n a_{z,v}(\theta_a) \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_a)$$

which is represented in SMF as

$$p(X^t, \mathbf{y}_e^t)(\theta_a) = O^t A(\theta_a)^\top p(X^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_a)$$

The overall procedure of computing the sensitivity coefficients of $p(X^t, \mathbf{y}_e^t)(\theta_a)$ using the SMF is described in the following algorithm.

3.1.3. Algorithm Implementation

This algorithm summarizes the proposed technique, where e is the sequence of observations and $\theta = a_{r,s}$, and solves the recursive Equation (7). Once the Transition and Observation matrices are represented in SMF as explained above, the sensitivity coefficients are computed by filling the forward matrices F^k for $k = 1 \cdots t$. The details of filling contents of the forward matrices in Algorithm 1 are explained as follows. F^1 is initialized by the matrix multiplications of O^1 , which is a diagonal matrix whose v^{th} diagonal entry is $P(y_e^1|x_v^1)$, and the initial distribution vector Γ^\top ,

$$F^1 = O^1 * \Gamma^\top$$

The remaining matrices F^k of size $n \times k, 2 \leq k \leq t$, are initialized by filling them with zeros. Two temporary matrices $Ftmp_1$ and $Ftmp_2$ are used for computing the coefficients as explained previously. $Ftmp_1$ is computed as the matrix multiplication of O^k, \bar{A}^\top (which is independent of θ_a) and F^{k-1} , and a zero vector of size n (= the number of states) is appended at its end. F^{k-1} represents the recursion in the filtering probability. In the same way $Ftmp_2$ is computed as the matrix multiplication

of O^k , \widehat{A}^\top (which is dependent on θ_a) and F^{k-1} , and a zero vector of size n is appended to the front. Then F^k is set as the sum of $Ftmp_1$ and $Ftmp_2$:

$$F^k = Ftmp_1 + Ftmp_2$$

Algorithm 1 Computes the coefficients of the forward (filtering) probability sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_a)$ with $\theta_a = a_{r,s}$ in forward matrices F^k for $k = 1, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_a

Output: F^1, \dots, F^t

- 1: $\bar{A} \leftarrow A$
 - 2: $\bar{A}_{r,s} \leftarrow 0$
 - 3: $\bar{A}_{r,:} \leftarrow \bar{A}_{r,:} / (1 - a_{r,s})$
 - 4: $\widehat{A} \leftarrow A - \bar{A}$
 - 5: $\widehat{A}_{r,:} \leftarrow \widehat{A}_{r,:} / a_{r,s}$
 - 6: $O^1 \leftarrow \text{diag}(O, \mathbf{e}^1)$
 - 7: $F^1 \leftarrow O^1 * \Gamma^\top$
 - 8: **for** $k = 2$ to t **do**
 - 9: $O^k \leftarrow \text{diag}(O, \mathbf{e}^k)$
 - 10: $Ftmp_1 \leftarrow [O^k * \bar{A}^\top * F^{k-1}, \text{zeros}(n, 1)]$
 - 11: $Ftmp_2 \leftarrow [\text{zeros}(n, 1), O^k * \widehat{A}^\top * F^{k-1}]$
 - 12: $F^k \leftarrow Ftmp_1 + Ftmp_2$
 - 13: **Return** F^1, \dots, F^t
-

3.2. Initial Parameter Variation

In this subsection, the sensitivity of the filtering probability to the initial parameter variation is derived based on SMF. Let us consider the sensitivity functions $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma)$ for a filtering probability and variation in the initial parameter $\theta_\gamma = \gamma_r$. From Equation (6), the recursive filtering probability sensitivity expression, for $t = 1$, we have $p(x_v^1, \mathbf{y}_e^1)(\theta_\gamma) = o_{v,\mathbf{e}^1} \cdot \gamma_v(\theta_\gamma)$, and for $t > 1$,

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma) = o_{v,\mathbf{e}^t} \cdot \sum_{z=1}^n a_{z,v} \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_\gamma) \tag{8}$$

3.2.1. Initial Vector Decomposition

In the proposed algorithm, to calculate the coefficients in a series of forward matrices $F^k, k = 1, \dots, t$, the Initial Vector Γ is divided into $\bar{\Gamma}$ and $\widehat{\Gamma}$ that are independent of and dependent on θ_γ , respectively, for mathematical convenience. The matrices $\bar{\Gamma}$ and $\widehat{\Gamma}$ are computed based on the proportional co-variation assumption shown in Equation (4).

Let us explain the Initial Vector Γ decomposition considering the HMM presented in Section 3.1 where the Initial Vector Γ is $[\gamma_1, \gamma_2, \dots, \gamma_n]$ and the initial parameter variation θ_γ is $\gamma_2 = p(x_2^1)$. This makes the Initial Vector Γ a function of θ_γ , and, with the proportional co-variation assumption shown in Equation (4), it becomes

$$\Gamma(\theta_\gamma) = \left[\frac{\gamma_1(1-\theta_\gamma)}{1-\gamma_2} \quad \theta_\gamma \quad \dots \quad \frac{\gamma_n(1-\theta_\gamma)}{1-\gamma_2} \right]$$

This matrix is divided into $\bar{\Gamma}$ and $\widehat{\Gamma}$ that are independent of and dependent on θ_γ , respectively, as follows, and such that $\Gamma(\theta_\gamma) = \bar{\Gamma} + \widehat{\Gamma}(\theta_\gamma)$.

$$\bar{\Gamma} = \left[\frac{\gamma_1}{1-\gamma_2} \quad 0 \quad \dots \quad \frac{\gamma_n}{1-\gamma_2} \right] \quad \text{and} \quad \widehat{\Gamma}(\theta_\gamma) = \left[\frac{-\gamma_1}{1-\gamma_2} \quad 1 \quad \dots \quad \frac{-\gamma_n}{1-\gamma_2} \right] \cdot \theta_\gamma$$

In short, this decomposition can be represented simply as $\bar{\Gamma}$ and $\hat{\Gamma}$ that are independent of and dependent on θ_γ , respectively.

Once the Initial Vector Γ is decomposed as shown above, to compute the coefficients of the sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma)$ using the recursive Equation (8), the Observation Matrix O is represented as an $n \times n$ diagonal matrices O^t .

3.2.2. SMF for Sensitivity Analysis of the Filtering Probability to the Initial Parameter Variation

In the simplified matrix representation, the sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma)$ shown in Equation (8) can be written as a simple matrix-vector multiplication. It follows that for $t = 1$ we have the constant $p(x_v^1, \mathbf{y}_e^1)(\theta_\gamma) = o_{v,e^1} \cdot \gamma_v(\theta_\gamma)$, which becomes $p(X^1, \mathbf{y}_e^1)(\theta_\gamma) = O^1 \Gamma^T(\theta_\gamma)$ in the simplified matrix representation. For $t > 1$, we have

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma) = o_{v,e^t} \cdot \sum_{z=1}^n a_{z,v} \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_\gamma)$$

which is represented as

$$p(X^t, \mathbf{y}_e^{1:t})(\theta_\gamma) = O^t A^T p(X^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_\gamma)$$

To solve the forward (filtering) probability sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma)$, the following algorithm implementation is presented.

3.2.3. Algorithm Implementation

This procedure summarizes the proposed method, where e is the sequence of observations and $\theta = \theta_\gamma$, and solves the recursive Equation (8).

As shown in Algorithm 2, once the Initial Vector is decomposed into the parts, $\bar{\Gamma}$ independent of θ_γ and $\hat{\Gamma}$ dependent on θ_γ and the Observation Matrix is represented in the diagonal matrices, the sensitivity coefficients are computed by filling the forward matrices F^k for $k = 1 \dots t$. The details of filling the forward matrices in Algorithm 2 are discussed as follows. F^1 is initialized as a sum of two temporary matrices $Ftmp_1$ and $Ftmp_2$ which are used to compute the coefficients for the parts independent of and dependent on θ_γ , respectively. $Ftmp_1$ is computed as the matrix multiplication of O^1 and $\bar{\Gamma}'$, and a zero vector of size n is appended at its end to represent the zero coefficients of degree 1. In the same way, $Ftmp_2$ is computed as the matrix multiplication of O^1 and $\hat{\Gamma}'$, and a zero vector of size n is appended in front of it to represent the zero coefficients of degree 0. Then

Algorithm 2 This computes the coefficients of the forward (filtering) probability sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_\gamma)$ with $\theta_\gamma = \gamma_r$ in forward matrices F^k for $k = 1, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_γ

Output: F^1, \dots, F^t

- 1: $\bar{\Gamma} \leftarrow \Gamma$
 - 2: $\bar{\Gamma}_r \leftarrow 0$
 - 3: $\bar{\Gamma} \leftarrow \bar{\Gamma} ./ (1 - \gamma_r)$
 - 4: $\hat{\Gamma} \leftarrow \Gamma - \bar{\Gamma}$
 - 5: $\hat{\Gamma} \leftarrow \hat{\Gamma} ./ \gamma_r$
 - 6: $O^1 \leftarrow \text{diag}(O_{:,e^1})$
 - 7: $Ftmp_1 \leftarrow [O^1 * \bar{\Gamma}', \text{zeros}(n, 1)]$
 - 8: $Ftmp_2 \leftarrow [\text{zeros}(n, 1), O^1 * \hat{\Gamma}']$
 - 9: $F^1 \leftarrow Ftmp_1 + Ftmp_2$
 - 10: **for** $k = 2$ **to** t **do**
 - 11: $O^k \leftarrow \text{diag}(O_{:,e^k})$
 - 12: $F^k \leftarrow [O^k * A' * F^{k-1}]$
 - 13: **Return** F^1, \dots, F^t
-

$$F^1 = Ftmp_1 + Ftmp_2$$

The remaining matrices F^k of size $n \times k, 2 \leq k \leq t$, are computed using recursion. They are computed as the matrix multiplication of O^k, A' and F^{k-1} .

3.3. Observation Parameter Variation

In this subsection, the sensitivity of the filtering probability $p(x_v^t, \mathbf{y}_e^{1:t})$ to the observation parameter variation is derived based on SMF. Let us consider the sensitivity functions $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_o)$ for a filtering probability and variation in the observation parameter $\theta_o = o_{r,s}$. From Equation (6), the recursive filtering probability sensitivity function, it follows that for $t = 1$ we have $p(x_v^1, \mathbf{y}_e^1)(\theta_o) = o_{v,e^1}(\theta_o) \cdot \gamma_v$, and for $t > 1$,

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_o) = o_{v,e^t}(\theta_o) \cdot \sum_{z=1}^n a_{z,v} \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_o) \tag{9}$$

3.3.1. Observation Matrix Decomposition

In the proposed algorithm, to calculate the coefficients in a series of forward matrices $F^k, k = 1, \dots, t$, the Observation Matrix O is divided into \bar{O} and \hat{O} that are independent of and dependent on θ_o , respectively, for mathematical convenience. The matrices \bar{O} and \hat{O} are computed based on the proportional co-variation assumed as shown in Equation (4). Let us explain the Observation Matrix O decomposition by considering the HMM presented in the previous section and let the observation parameter θ_o be $o_{2,1} = p(y_1^t | x_2^t)$.

The observation matrix O becomes a function of θ_o based on the proportional co-variation assumption shown in Equation (4) as follows.

$$O(\theta_o) = \begin{bmatrix} o_{1,1} & o_{1,2} & \cdots & o_{1,m} \\ \theta_o & \frac{o_{2,2}(1-\theta_o)}{1-o_{2,1}} & \cdots & \frac{o_{2,m}(1-\theta_o)}{1-o_{2,1}} \\ \vdots & \vdots & \ddots & \vdots \\ o_{n,1} & o_{n,2} & \cdots & o_{n,m} \end{bmatrix}$$

This matrix is decomposed into \bar{O} and \hat{O} that are independent of and dependent on θ_o , respectively, such that $O(\theta_o) = \bar{O} + \hat{O}(\theta_o)$, just as we decomposed $A(\theta_a) = \bar{A} + \hat{A}(\theta_a)$ in Section 3.1.1.

Once the Observation Matrix is decomposed into \bar{O} and \hat{O} , in the SMF, they are represented as $n \times n$ diagonal matrices. For example, the sequence of observations y_2^1, y_1^2 and y_3^3 , the corresponding diagonal matrices become

$$\begin{aligned} \bar{O}^1 &= \text{diag}(\bar{O}_{:,2}), & \bar{O}^2 &= \text{diag}(\bar{O}_{:,1}) & \text{and} & \bar{O}^3 &= \text{diag}(\bar{O}_{:,3}) \\ \hat{O}^1 &= \text{diag}(\hat{O}_{:,2}), & \hat{O}^2 &= \text{diag}(\hat{O}_{:,1}) & \text{and} & \hat{O}^3 &= \text{diag}(\hat{O}_{:,3}) \end{aligned}$$

3.3.2. SMF for Sensitivity Analysis of the Filtering Probability to the Observation Parameter Variation

In the simplified matrix representation, the sensitivity function $p(x_v^t, \mathbf{y}_e^{1:t})(\theta_o)$ can be written as a simple matrix-vector multiplication. From Equation (9), it follows that for $t = 1$ we have the constant $p(x_v^1, \mathbf{y}_e^1)(\theta_o) = o_{v,e^1}(\theta_o) \cdot \gamma_v$, which becomes $p(x_v^1, \mathbf{y}_e^1)(\theta_o) = O^1(\theta_o)\Gamma$ in the SMF. For $t > 1$, we have

$$p(x_v^t, \mathbf{y}_e^{1:t})(\theta_o) = o_{v,e^t}(\theta_o) \cdot \sum_{z=1}^n a_{z,v} \cdot p(x_v^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_o),$$

which is represented as

$$p(X^t, \mathbf{y}_e^{1:t})(\theta_o) = O^t(\theta_o)A^\top p(X^{t-1}, \mathbf{y}_e^{1:t-1})(\theta_o)$$

The derivation of the proposed technique for the sensitivity analysis of the forward (filtering) probability $p(x_o^t, \mathbf{y}_e^{1:t}) (\theta_o)$ with observation parameter variation $\theta_o = o_{r,s}$ is formulated in Algorithm 3.

Algorithm 3 This computes the coefficients of the forward (filtering) probability sensitivity function $p(x_o^t, \mathbf{y}_e^{1:t}) (\theta_o)$ with $\theta_o = o_{r,s}$ in forward matrices F^k for $k = 1, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_o

Output: F^1, \dots, F^t

```

1:  $\bar{O} \leftarrow O$ 
2:  $\bar{O}_{r,s} \leftarrow 0$ 
3:  $\bar{O}_{r,:} \leftarrow \bar{O}_{r,:} / (1 - o_{r,s})$ 
4:  $\hat{O} \leftarrow O - \bar{O}$ 
5:  $\hat{O}_{r,:} \leftarrow \hat{O}_{r,:} / o_{r,s}$ 
6:  $\bar{O}^1 \leftarrow \text{diag}(\bar{O}_{:,e^1})$ 
7:  $\hat{O}^1 \leftarrow \text{diag}(\hat{O}_{:,e^1})$ 
8:  $Ftmp_1 \leftarrow [\bar{O}^1 * \Gamma', \text{zeros}(n, 1)]$ 
9:  $Ftmp_2 \leftarrow [\text{zeros}(n, 1), \hat{O}^1 * \Gamma']$ 
10:  $F^1 \leftarrow Ftmp_1 + Ftmp_2$ 
11: for  $k = 2$  to  $t$  do
12:    $\bar{O}^k \leftarrow \text{diag}(\bar{O}_{:,e^k})$ 
13:    $\hat{O}^k \leftarrow \text{diag}(\hat{O}_{:,e^k})$ 
14:    $Ftmp_1 \leftarrow [\bar{O}^k * A' * F^{k-1}, \text{zeros}(n, 1)]$ 
15:    $Ftmp_2 \leftarrow [\text{zeros}(n, 1), \hat{O}^k * A' * F^{k-1}]$ 
16:    $F^k \leftarrow Ftmp_1 + Ftmp_2$ 
17: Return  $F^1, \dots, F^t$ 

```

3.3.3. Algorithm Implementation

In this algorithm, e is the sequence of observation and $\theta = \theta_o$. It solves the recursive sensitivity function Equation (9) in forward matrices F^k for $k = 1, \dots, t$. The observation matrix is decomposed into the independent \bar{O} and dependent \hat{O} observation matrices on θ_o . Then, these observation matrices are represented as diagonal matrices in the SMF computation. The detailed steps of filling the contents of the forward matrices F^k are explained as follows. F^1 is initialized as a sum of two temporary matrices $Ftmp_1$ and $Ftmp_2$, which are used to compute the coefficients for the independent and dependent parts on θ_o . $Ftmp_1$ is computed as the matrix multiplication of \bar{O}^1 and Γ' , and a zero vector of size n states is appended at its end to represent the zero coefficients of degree 1. In the same way, $Ftmp_2$ is computed as the matrix multiplication of \hat{O}^1 and Γ' , and a zero vector of size n states is appended in front of it to represent the zero coefficients of degree 0.

$$F^1 = Ftmp_1 + Ftmp_2$$

The remaining matrices F^k of size $n \times k, 2 \leq k \leq t$, are computed using recursion. Again, two temporary matrices $Ftmp_1$ and $Ftmp_2$ are used for computing the coefficients as it is explained above. Here, $Ftmp_1$ is computed as the matrix multiplication of the diagonal matrix \bar{O}^k that is independent of θ_o, A' and F^{k-1} , and a zero vector of size n states is appended at its end. F^{k-1} represents the recursion in the filtering probability. Likewise, $Ftmp_2$ is computed as the matrix multiplication of the diagonal matrix \hat{O}^k that is dependent on θ_o, A' and F^{k-1} , and a zero vector of size n states is appended in front of it. Then, F^k is set as the sum of $Ftmp_1$ and $Ftmp_2$:

$$F^k = Ftmp_1 + Ftmp_2$$

3.4. Filtering Sensitivity Function Example

The proposed algorithm is illustrated by the following example. The example is also used to illustrate the computations in the Coefficient-Matrix-Fill procedure in [26]. The computed coefficients using the proposed algorithm are the same as those computed using the Coefficient-Matrix-Fill procedure. However, the proposed algorithm is simplified and works for any number of hidden states and evidence (observation) variables. It is also efficient in terms of computational time as the length of the observation sequence increases.

Example 1. Consider an HMM with binary-valued hidden state X and binary-valued evidence variable Y . Let $\Gamma = [0.20, 0.80]$ be the initial vector for X^1 . The transition matrix A and observation matrix O be as follows:

$$A = \begin{bmatrix} 0.95 & 0.05 \\ 0.15 & 0.85 \end{bmatrix} \quad \text{and} \quad O = \begin{bmatrix} 0.75 & 0.25 \\ 0.90 & 0.10 \end{bmatrix}$$

Let us compute the sensitivity functions for the filtering probability using Algorithm 1 for the two states of X^t as a function of transition parameter $\theta_a = a_{2,1} = p(x_1^t | x_2^{t-1}) = 0.15$ given the following sequence of observations: y_2^1, y_1^2 and y_1^3 .

The simplified matrix formulation procedure divides the Transition Matrix A into \bar{A} and \hat{A} :

$$A = \begin{bmatrix} 0.95 & 0.05 \\ \theta_a & 1 - \theta_a \end{bmatrix}, \quad \bar{A} = \begin{bmatrix} 0.95 & 0.05 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad \hat{A} = \begin{bmatrix} 0 & 0 \\ 1 & -1 \end{bmatrix}$$

For the observation sequence given, y_2^1, y_1^2 and y_1^3 , the diagonal matrices become

$$O^1 = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.10 \end{bmatrix}, \quad O^2 = \begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix} \quad \text{and} \quad O^3 = \begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix}$$

The coefficients for the sensitivity functions are computed by the proposed procedure using the forward matrices that are constructed as follows:

$$F^1 = O^1 * \Gamma' = \begin{bmatrix} 0.25 & 0 \\ 0 & 0.10 \end{bmatrix} \times \begin{bmatrix} 0.20 \\ 0.80 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 0.08 \end{bmatrix}$$

Then, $F^2 = Ftmp_1 + Ftmp_2$, where the temporary matrices $Ftmp_1$ and $Ftmp_2$ are computed as follows:

$$\begin{aligned} Ftmp_1 &= [O^2 * \bar{A}' * F^1, \text{zeros}(2,1)] \\ &= \left[\begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix} \times \begin{bmatrix} 0.95 & 0 \\ 0.05 & 1 \end{bmatrix} \times \begin{bmatrix} 0.05 \\ 0.08 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right] \\ &= \begin{bmatrix} 0.0356 & 0 \\ 0.0743 & 0 \end{bmatrix} \\ Ftmp_2 &= [\text{zeros}(2,1), O^2 * \hat{A}' * F^1] \\ &= \left[\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix} \times \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix} \times \begin{bmatrix} 0.05 \\ 0.08 \end{bmatrix} \right] \\ &= \begin{bmatrix} 0 & 0.060 \\ 0 & -0.072 \end{bmatrix} \end{aligned}$$

$$F^2 = Ftmp_1 + Ftmp_2 = \begin{bmatrix} 0.0356 & 0.060 \\ 0.0743 & -0.072 \end{bmatrix},$$

and finally, for F^3 :

$$\begin{aligned} Ftmp_1 &= [O^3 * \bar{A}' * F^2, \text{zeros}(2,1)] \\ &= \left[\left[\begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix} \times \begin{bmatrix} 0.95 & 0 \\ 0.05 & 1 \end{bmatrix} \times \begin{bmatrix} 0.0356 & 0.060 \\ 0.0743 & -0.072 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right] \\ &= \begin{bmatrix} 0.0254 & 0.0428 & 0 \\ 0.0684 & -0.0621 & 0 \end{bmatrix} \\ Ftmp_2 &= [\text{zeros}(2,1), O^3 * \hat{A}' * F^2] \\ &= \left[\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.75 & 0 \\ 0 & 0.90 \end{bmatrix} \times \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix} \times \begin{bmatrix} 0.0356 & 0.060 \\ 0.0743 & -0.072 \end{bmatrix} \right] \\ &= \begin{bmatrix} 0 & 0.0557 & -0.0540 \\ 0 & -0.0668 & 0.0648 \end{bmatrix} \\ F^3 &= Ftmp_1 + Ftmp_2 = \begin{bmatrix} 0.0254 & 0.0984 & -0.0540 \\ 0.0684 & -0.1289 & 0.0648 \end{bmatrix} \end{aligned}$$

From F^3 , we have the sensitivity function

$$p(x_1^3, \mathbf{y}_e^{1:3})(\theta_a) = 0.0254 + 0.0984 \cdot \theta_a - 0.054 \cdot \theta_a^2,$$

and from F^2 ,

$$p(x_2^2, \mathbf{y}_e^{1:2})(\theta_a) = 0.0743 - 0.072 \cdot \theta_a$$

The sensitivity functions for probability of evidence by summing column entries of the forward matrices become:

$$p(\mathbf{y}_e^{1:3})(\theta_a) = (F_{1,1}^3 + F_{2,1}^3) + (F_{1,2}^3 + F_{2,2}^3) \cdot \theta_a + (F_{1,3}^3 + F_{2,3}^3) \cdot \theta_a^2,$$

and

$$p(\mathbf{y}_e^{1:2})(\theta_a) = (F_{1,1}^2 + F_{2,1}^2) + (F_{1,2}^2 + F_{2,2}^2) \cdot \theta_a$$

The sensitivity functions for the two filtering tasks become:

$$p(x_1^3 | \mathbf{y}_e^{1:3})(\theta_a) = \frac{-0.054 \cdot \theta_a^2 + 0.0984 \cdot \theta_a + 0.0254}{0.0108 \cdot \theta_a^2 - 0.0305 \cdot \theta_a + 0.0938},$$

and

$$p(x_2^2 | \mathbf{y}_e^{1:2})(\theta_a) = \frac{-0.072 \cdot \theta_a + 0.0743}{-0.012 \cdot \theta_a + 0.1099},$$

these sensitivity functions are displayed in Figure 2.

The derivative of these sensitivity functions can be used to analyze the sensitive of the filtering tasks for the change in parameter θ_a .

$$\frac{dp(x_1^3 | \mathbf{y}_e^{1:3})(\theta_a)}{d\theta_a} = \frac{-0.006 \cdot \theta_a^2 - 0.0107 \cdot \theta_a - 0.0094}{(0.0108 \cdot \theta_a^2 - 0.0305 \cdot \theta_a + 0.0938)^2},$$

and

$$\frac{dp(x_2^2|y_e^{1:2})(\theta_a)}{d\theta_a} = \frac{-0.0070}{(-0.012 \cdot \theta_a + 0.1099)^2}$$

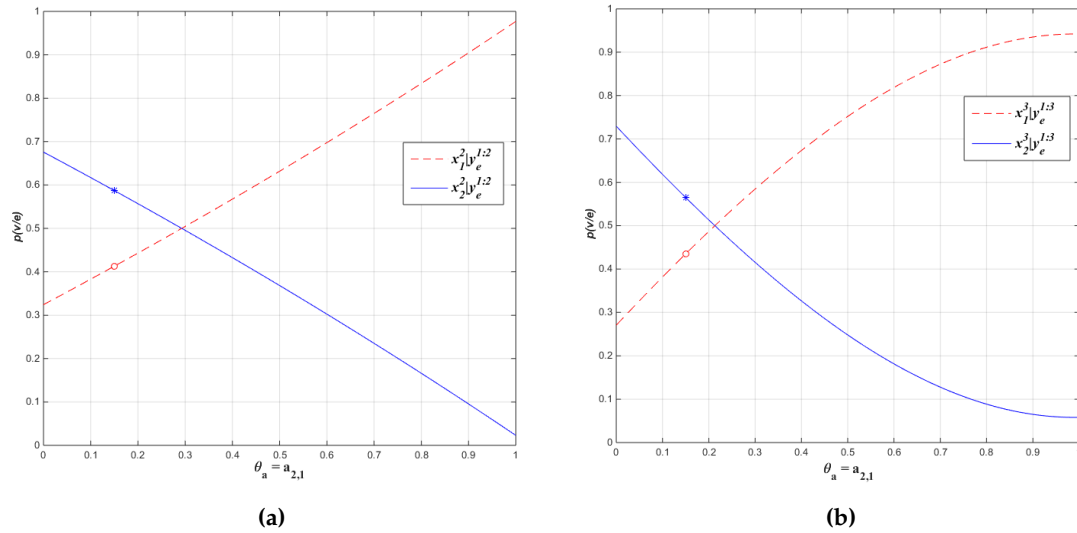


Figure 2. Sensitivity functions: (a) $p(X^2|y_e^{1:2})(\theta_a)$ for both states of X^2 . (b) $p(X^3|y_e^{1:3})(\theta_a)$ for both states of X^3 .

3.5. Run-time Analysis

The time for computing the coefficients of the filtering probabilities for transition parameter variation is recorded for the Coefficient-Matrix-Fill procedure and our method (SMF) using the HMM shown in Example (1). It is shown in Figure 3, where the Coefficient-Matrix-Fill procedure is labeled as CMFP. In [26], the algorithm for the Coefficient-Matrix-Fill procedure is presented. It is shown for an HMM model with $n = 2$ hidden states and $m = 2$ observable symbols. The proposed SMF algorithm has shown a significant improvement in computational time compared to the Coefficient-Matrix-Fill procedure as the length of the observation sequence increases. For example, for an observation sequence length of 1000, the proposed technique takes a mean time of 0.048 s. In comparison, it takes the Coefficient-Matrix-Fill procedure 2.183 seconds. This is an improvement of 98%. The algorithms are implemented in MATLAB on a 64-bit Windows 7 Professional Workstation (Dell Precision T7600 with Intel(R) Xeon CPU E5-2680 0 @ 2.70 GHz dual core processors and 64GB RAM).

The computational complexity of our method is linear time, $O(l)$, where l is the length of the observation sequence, whereas the Coefficient-Matrix-Fill procedure is quadratic time, $O(l^2)$. In our method, the sensitivity function coefficients are computed in *one for loop* that runs for the length of the observation sequence, as shown in Algorithms 1–6, whereas in the Coefficient-Matrix-Fill procedure *two nested for loops* are used. It computes the sensitivity coefficients in the forward matrices element-by-element in the *inner loop*, which runs in increasing time up to the maximum of the length of the observation sequence, and the *outer loop* runs for the length of the observation sequence.

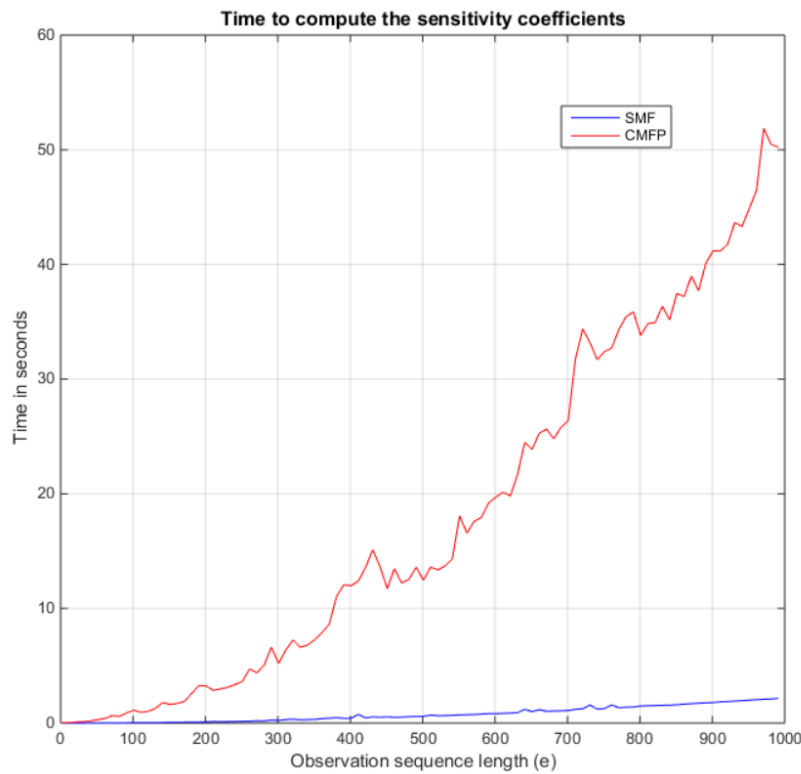


Figure 3. Time in seconds to compute the sensitivity coefficients for an observation sequence length from 1 to 1000 with a step size of 10.

4. Sensitivity Analysis of Smoothing Probability

In this section, the sensitivity analysis of smoothing probability for variation of the transition, initial and observation parameters are presented using simplified matrix formulation. The sensitivity function $p(x_v^t, \mathbf{y}_e^{1:T})(\theta)$ where $t < T$ and θ is one of the model parameters, can be computed using the recursive probability expression shown in Equation (2) and (3).

$$p(x_v^t, \mathbf{y}_e^{1:T})(\theta) = p(x_v^t, \mathbf{y}_e^{1:t}, \mathbf{y}_e^{t+1:T})(\theta) = p(\mathbf{y}_e^{t+1:T} | x_v^t)(\theta) \cdot p(x_v^t, \mathbf{y}_e^{1:t})(\theta)$$

The first term in this product can be computed using the backward procedure as shown in Equation (3); the second term is the filtering probability that is computed using the forward procedure. Consequently, the sensitivity function for the smoothing probability $p(x_v^t, \mathbf{y}_e^{1:T})(\theta)$ is the product of the polynomial sensitivity functions computed using the forward and backward procedures. The sensitivity function for the filtering probability $p(x_v^t, \mathbf{y}_e^{1:t})(\theta)$ is discussed in Section 3. From Equation (3), the backward recursive probability expression, it follows that

$$p(\mathbf{y}_e^{t+1:T} | x_v^t)(\theta) = \begin{cases} \sum_{z=1}^n o_{z,e^T}(\theta) \cdot a_{v,z}(\theta) & \text{if } t = T - 1 \\ \sum_{z=1}^n o_{z,e^{t+1}}(\theta) \cdot a_{v,z}(\theta) \cdot p(\mathbf{y}_e^{t+2:T} | x_z^{t+1})(\theta) & \text{if } t < T - 1, \end{cases} \tag{10}$$

where

$$a_{v,z}(\theta) = \begin{cases} a_{v,z}(\theta_a) & \text{for transition parameter variation } \theta = \theta_a \\ a_{v,z} & \text{otherwise,} \end{cases}$$

and

$$o_{z,e^t}(\theta) = \begin{cases} o_{z,e^t}(\theta_0) & \text{for observation parameter variation } \theta = \theta_0 \\ o_{z,e^t} & \text{otherwise} \end{cases}$$

To compute the coefficients of the sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta)$, a series of Backward matrices $B^k, k = t, \dots, T - 1$, based on the simplified matrix formulation as presented in Section 3 are used. In the next subsections, the sensitivity functions for the smoothing probability $p(x_v^t, \mathbf{y}_e^{1:T})(\theta)$ for transition, initial and observation parameter variations are discussed.

4.1. Transition Parameter Variation

In the simplified matrix formulation, the sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_a)$ for transition parameter $\theta_a = a_{r,s}$ can be represented as follows:

$$p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_a) = A(\theta_a)O^{t+1}p(\mathbf{y}_e^{t+2:T}|x_z^{t+1})(\theta_a)$$

For $t = T - 1, p(\mathbf{y}_e^{T+1:T}|x_z^T)(\theta_a) = 1$, the recursive probability expression reduces to the following:

$$p(\mathbf{y}_e^{T:T}|x_v^{T-1})(\theta_a) = A(\theta_a)O^{t+1}$$

The proposed technique for the sensitivity analysis of the backward probability sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_a)$ for the transition parameter is formulated in Algorithm 4.

Algorithm 4 This computes the coefficients of the backward probability sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_a)$ in backward matrices B^k for $k = T, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_a

Output: B^T, \dots, B^t

- 1: $\bar{A} \leftarrow A$
 - 2: $\bar{A}_{r,s} \leftarrow 0$
 - 3: $\bar{A}_{r,:} \leftarrow \bar{A}_{r,:} / (1 - a_{r,s})$
 - 4: $\hat{A} \leftarrow A - \bar{A}$
 - 5: $\hat{A}_{r,:} \leftarrow \hat{A}_{r,:} / a_{r,s}$
 - 6: $B^T \leftarrow \text{ones}(n, 1)$
 - 7: **for** $k = T - 1$ to t **do**
 - 8: $O^{k+1} \leftarrow \text{diag}(O_{:,e^{k+1}})$
 - 9: $Btmp_1 \leftarrow [\bar{A} * O^{k+1} * B^{k+1}, \text{zeros}(n, 1)]$
 - 10: $Btmp_2 \leftarrow [\text{zeros}(n, 1), \hat{A} * O^{k+1} * B^{k+1}]$
 - 11: $B^k \leftarrow Btmp_1 + Btmp_2$
 - 12: **Return** B^T, \dots, B^t
-

4.1.1. Algorithm Implementation

In this algorithm, e is the sequence of observation and $\theta = \theta_a$. It solves the recursive backward probability sensitivity function Equation (10) using the backward matrices B^k for $k = T, \dots, t$. Once the transition matrix is decomposed into the transition matrices \bar{A} independent of θ_a and \hat{A} dependent on θ_a , the coefficients of the sensitivity function are computed by filling the backward matrices B^k as follows. First, B^T is initialized as a vector of ones for n hidden states.

$$B^T = \text{ones}(n, 1)$$

The remaining matrices B^k of size $n \times k, k = T - 1 \leq k \leq t$ are computed by filling them as follows: Once the observation matrix O^{k+1} is represented as a diagonal matrix, two temporary matrices $Btmp_1$ and $Btmp_2$ are used for computing the coefficients as previously explained. $Btmp_1$ is computed

as the matrix multiplication of \bar{A} that is independent of θ_a , O^{k+1} and B^{k+1} and a zero vector of size n is appended at its end. B^{k+1} represents the recursion in the backward probability. The same way $Btmp_2$ is computed as the matrix multiplication of \hat{A} that is dependent on θ_a , O^{k+1} and B^{k+1} and a zero vector of size n is appended to the front. Then B^k is set as the sum of $Btmp_1$ and $Btmp_2$,

$$B^k = Btmp_1 + Btmp_2$$

4.2. Initial Parameter Variation

In this subsection, the sensitivity of the backward probability to initial parameter variation is derived based on SMF. The sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_\gamma)$ for initial parameter variation $\theta_\gamma = \gamma_r$, as shown in the recursive backward probability sensitivity function in Equation (10), can be represented in SMF as

$$p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_\gamma) = AO^{t+1}p(\mathbf{y}_e^{t+2:T}|x_z^{t+1})(\theta_\gamma)$$

For $t = T - 1$, $p(\mathbf{y}_e^{T+1:T}|x_z^T)(\theta_\gamma) = 1$ and the recursive probability expression reduces to

$$p(\mathbf{y}_e^{T:T}|x_v^{T-1})(\theta_\gamma) = AO^{t+1}$$

The procedure to compute the coefficients of the function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_\gamma)$ in backward matrices B^k for $k = T, \dots, t$ using SMF is summarized in Algorithm 5.

Algorithm 5 This computes the coefficients of the backward probability sensitivity function $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_\gamma)$ with $\theta_\gamma = \gamma_r$ in backward matrices B^k for $k = T, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_γ

Output: B^T, \dots, B^t

- 1: $B^T \leftarrow \text{ones}(n, 1)$
 - 2: **for** $k = T - 1$ **to** t **do**
 - 3: $O^{k+1} \leftarrow \text{diag}(O_{:, \mathbf{e}^{k+1}})$
 - 4: $B^k \leftarrow A * O^{k+1} * B^{k+1}$
 - 5: **Return** B^T, \dots, B^t
-

4.2.1. Algorithm Implementation

As discussed above, the sensitivity function for the backward probability $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_\gamma)$ with $\theta_\gamma = \gamma_r$ is a polynomial of degree zero. The contents of the backward matrices B^k are filled as follows. B^T is initialized as a vector of ones for n hidden states.

$$B^T = \text{ones}(n, 1)$$

The remaining matrices B^k of size $n \times k, k = T - 1 \leq k \leq t$, are computed by filling them as follows. Once the observation matrix O^{k+1} is represented as a diagonal matrix, B^k is set as the matrix multiplication of A, O^{k+1} and B^{k+1} . B^{k+1} represents the recursion in the backward probability.

4.3. Observation Parameter Variation

The sensitivity function for the backward probability $p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_o)$ for the observation parameter variation $\theta_o = o_{r,s}$ can be represented as follows:

$$p(\mathbf{y}_e^{t+1:T}|x_v^t)(\theta_o) = AO^{t+1}(\theta_o)p(\mathbf{y}_e^{t+2:T}|x_z^{t+1})(\theta_o)$$

For $t = T - 1$, $p(\mathbf{y}_e^{T+1:T}|x_z^T)(\theta_o) = 1$, the recursive probability expression reduces to the following

$$p(\mathbf{y}_e^{T:T}|x_v^{T-1})(\theta_o) = AO^{t+1}(\theta_o)$$

The sensitivity analysis of the backward probability sensitivity function $p(\mathbf{y}_e^{t+1:T} | x_v^t)(\theta_o)$ for observation parameter variation $\theta_o = o_{r,s}$ is summarized in Algorithm 6.

Algorithm 6 This computes the coefficients of the backward probability sensitivity function $p(\mathbf{y}_e^{t+1:T} | x_v^t)(\theta_o)$ with $\theta_o = o_{r,s}$ in backward matrices B^k for $k = T, \dots, t$ using SMF.

Input: A, O, Γ, \mathbf{e} , and θ_o

Output: B^T, \dots, B^t

- 1: $\bar{O} \leftarrow O$
 - 2: $\bar{O}_{r,s} \leftarrow 0$
 - 3: $\bar{O}_{r,:} \leftarrow \bar{O}_{r,:} / (1 - o_{r,s})$
 - 4: $\hat{O} \leftarrow O - \bar{O}$
 - 5: $\hat{O}_{r,:} \leftarrow \hat{O}_{r,:} / o_{r,s}$
 - 6: $B^T \leftarrow \text{ones}(n, 1)$
 - 7: **for** $k = T - 1$ **to** t **do**
 - 8: $\bar{O}^{k+1} \leftarrow \text{diag}(\bar{O}_{:,e^{k+1}})$
 - 9: $\hat{O}^{k+1} \leftarrow \text{diag}(\hat{O}_{:,e^{k+1}})$
 - 10: $Btmp_1 \leftarrow [A * \bar{O}^{k+1} * B^{k+1}, \text{zeros}(n, 1)]$
 - 11: $Btmp_2 \leftarrow [\text{zeros}(n, 1), A * \hat{O}^{k+1} * B^{k+1}]$
 - 12: $B^k \leftarrow Btmp_1 + Btmp_2$
 - 13: **Return** B^T, \dots, B^t
-

4.3.1. Algorithm Implementation

In this algorithm, e is the sequence of observation and $\theta = \theta_o$ and it solves the recursive backward probability sensitivity function Equation (10) in the backward matrices B^k for $k = T, \dots, t$ using SMF. Once the observation matrix is decomposed into the independent \bar{O} and dependent \hat{O} observation matrices on θ_o , the coefficients of the sensitivity function $p(\mathbf{y}_e^{t+1:T} | x_v^t)(\theta_o)$ are computed by filling the backward matrices B^k as follows: B^T is initialized as a vector of ones for n hidden states.

$$B^T = \text{ones}(n, 1)$$

The remaining matrices B^k of size $n \times k, k = T - 1 \leq k \leq t$, are computed by filling them as follows: Once the independent \bar{O}^{k+1} and dependent \hat{O}^{k+1} observation matrices on θ_o is represented as a diagonal matrix, two temporary matrices $Btmp_1$ and $Btmp_2$ are used for computing the coefficients of the sensitivity function as previously explained. $Btmp_1$ is computed as the matrix multiplication of A, \bar{O}^{k+1} and B^{k+1} . A zero vector of size n is appended at its end. B^{k+1} represents the recursion in the backward probability. In the same way $Btmp_2$ is computed as the matrix multiplication of A, \hat{O}^{k+1} and B^{k+1} and a zero vector of size n is appended to the front. Then B^k is set as the sum of $Btmp_1$ and $Btmp_2$,

$$B^k = Btmp_1 + Btmp_2$$

5. Conclusions and Future Research

This research has shown that it is more efficient to compute the coefficients for the HMM sensitivity function directly from the HMM representation. The proposed method exploits the simplified matrix formulation for HMMs. A simple algorithm is presented which computes the coefficients for the sensitivity function of filtering and smoothing probabilities for transition, initial and observation parameter variation for all hidden states, as well as all time steps. This method differs from the other approaches in that it neither depends on a specific computational architecture nor requires a Bayesian network representation of the HMM.

The future extension of this work will include sensitivity analysis of predicted future observations $p(y_e^t | y_e^{1:T})(\theta), t > T$, and the most probable explanation for the corresponding parameter variations.

Future research on the sensitivity analysis of HMM, where different types of model parameters are varied simultaneously, as well as the case of continuous observations will be considered.

Acknowledgments: This work is partially supported by the US Department of Transportation (USDOT) under University Transportation Center (UTC) Program (DTRT13-G-UTC47), the Office of the Secretary of Defense (OSD) and the Air Force Research Laboratory under agreement number FA8750-15-2-0116. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes, notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of USDOT, OSD, Air Force Research Laboratory, or the U.S. Government. The first, and the second authors would like to acknowledge the support from the USDOT, and the second author would also like to acknowledge the support from the OSD and the Air Force.

Author Contributions: Seifemichael B. Amsalu and Abdollah Homaifar contributed equally to this work. Albert C. Esterline helped in writing the manuscript; All authors have read and approved the final manuscript.

Conflicts of Interest: The authors declare no conflicts of interest exist in this work.

References

- Patel, I.; Rao, Y.S. Speech recognition using hidden Markov model with MFCC-subband technique. *Recent Trends Inf. Telecommun. Comput.* **2010**, *5*, 168–172.
- Birney, E. Hidden Markov models in biological sequence analysis. *IBM J. Res. Dev.* **2001**, *45*, 449–454.
- Mamon, R.S.; Elliott, R.J. Hidden Markov models in finance. In *International Series in Operations Research & Management Science*, 2nd ed.; Springer: New York, NY, US, 2014. Available online: <https://link.springer.com/book/10.1007/978-1-4899-7442-6> (accessed on 11 August 2017).
- Bunke, H.; Caelli, T. *Hidden Markov Models: Applications in Computer Vision*; World Scientific Publishing: Danvers, MA, USA, 2001; pp. 1–223.
- Amsalu, S.B.; Homaifar, A.; Afghah, F.; Ramyar, S.; Kurt, A. Driver behavior modeling near intersections using support vector machines based on statistical feature extraction. *Intell. Veh. Symp.* **2015**, *8*, 1270–1275.
- Amsalu, S.B.; Homaifar, A. Driver behavior modeling near intersections using hidden Markov model based on genetic algorithm. *Intell. Transp. Eng.* **2016**, *10*, 193–200.
- Dymarski, P. *Hidden Markov Models, Theory and Applications*; InTech Open Access Publishers: Vienna, Austria, 2011; pp. 1–19. Available online: <https://cdn.intechopen.com/pdfs-wm/15359.pdf> (accessed on 11 August 2017).
- Murphy, K.P. Bayesian Networks: Representation, Inference and Learning. Ph.D. Thesis, University of California, Berkeley, CA, USA, 2002.
- Smyth, P.; Heckerman, D.; Jordan, M.I. Probabilistic independence networks for hidden Markov probability models. *Neural Comput.* **1997**, *9*, 227–269.
- Coquelin, P.A.; Deguest, R.; Munos, R. Sensitivity analysis in HMMs with application to likelihood maximization. *Adv. Neural Inf. Process. Syst.* **2009**, *22*, 387–395.
- Mitrophanov, A.Y.; Lomsadze, A.; Borodovsky, M. Sensitivity of hidden Markov models. *J. Appl. Probab.* **2005**, *42*, 632–642.
- Chan, H.; Darwiche, A. A distance measure for bounding probabilistic belief change. *Int. J. Approx. Reason.* **2005**, *38*, 149–174.
- Renooij, S.; Van der Gaag, L.C. Evidence-invariant sensitivity bounds. *Uncertain. Artif. Intell.* **2004**, *7*, 479–486.
- Renooij, S.; Van der Gaag, L.C. Exploiting evidence-dependent sensitivity bounds. *Uncertain. Artif. Intell.* **2012**. Available online: <https://arxiv.org/abs/1207.1357> (accessed on 11 August 2017).
- Renooij, S.; Van der Gaag, L.C. Evidence and scenario sensitivities in naive Bayesian classifiers. *Int. J. Approx. Reason.* **2008**, *49*, 398–416.
- Chan, H.; Darwiche, A. When do numbers really matter? *J. Artif. Intell. Res.* **2002**, *17*, 265–287.
- Van der Gaag, L.C.; Renooij, S.; Coupé, V.M.H. Sensitivity analysis of probabilistic networks. In *Advances in Probabilistic Graphical Models*; Springer: Berlin, Germany, 2007; pp. 103–124.
- Laskey, K.B. Sensitivity analysis for probability assessments in Bayesian networks. *IEEE Trans. Syst. Man Cybern.* **1995**, *25*, 901–909.

19. Jensen, F.V.; Nielsen, T.D. *Bayesian Networks and Decision Graphs*, 2nd ed.; Springer: Berlin, Germany, 2007; pp. 1–441.
20. Darwiche, A. A differential approach to inference in Bayesian networks. *J. Assoc. Comput. Mach.* **2003**, *50*, 280–305.
21. Coupé, V.M.H.; Jensen, F.V.; Kjærulff, U.; Van der Gaag, L.C. A computational architecture for N-way sensitivity analysis of Bayesian networks. *Dep. Comput. Sci.* **2000**. Available online: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.1434> (accessed on 11 August 2017).
22. Kjærulff, U.; Van der Gaag, L.C. Making sensitivity analysis computationally efficient. In *Uncertainty in Artificial Intelligence*; Morgan Kaufmann: San Francisco, CA, USA, 2000; pp. 317–325.
23. Chan, H.; Darwiche, A. Sensitivity analysis in Bayesian networks: From single to multiple parameters. In *Uncertainty in Artificial Intelligence*; AUAI Press: Arlington, VA, USA, 2004; pp. 67–75.
24. de Cooman, G.; Hermans, F.; Quaeghebeur, E. Sensitivity analysis for finite Markov chains in discrete time. *Uncertain. Artif. Intell.* **2008**, *8*, 129–136.
25. Renooij, S. Efficient sensitivity analysis in hidden Markov models. In *Proceedings of the Fifth European Workshop on Probabilistic Graphical Models*; HIIT Publications: Helsinki, Finland, 2010; pp. 241–248. Available online: <http://www.cs.uu.nl/research/techreps/repo/CS-2011/2011-025.pdf> (accessed on 19 August 2017).
26. Renooij, S. Efficient sensitivity analysis in hidden Markov models. *Int. J. Approx. Reason.* **2012**, *53*, 1397–1414.
27. Maua, D.; de Campos, C.; Antonucci, A. Algorithms for hidden Markov models with imprecisely specified parameters. *Br. Conf. Intell. Syst.* **2014**, 186–191, doi:10.1109/BRACIS.2014.42.
28. Baum, L.E.; Petrie, T. Statistical inference for probabilistic functions of finite state Markov chains. *Ann. Math. Stat.* **1966**, *37*, 1554–1563.
29. Rabiner, L.R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **1989**, *77*, 257–286.
30. Bilmes, J. A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. *Int. Comput. Sci. Inst.* **1998**, *4*, 126.
31. Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3rd ed.; Prentice Hall: New Jersey, NJ, USA, 2010; pp. 1–1154.
32. Charitos, T.; Van der Gaag, L.C. Sensitivity properties of Markovian models. In *Proceedings of Advances in Intelligent Systems—Theory and Applications Conference (AISTA)*; IEEE Computer Society: Luxembourg, 2004. Available online: <https://pdfs.semanticscholar.org/5d53/1a739b047fa6a126dfa32ba5cccd3f5934c.pdf> (accessed on 11 August 2017).
33. Van der Gaag, L.C.; Renooij, S. Analysing sensitivity data from probabilistic networks. *Uncertain. Artif. Intell.* **2001**, 530–537. Available online: <http://dl.acm.org/citation.cfm?id=2074087> (accessed on 11 August 2017).



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).