Journal of Artificial Intelligence Research 50 (2014) 189-233 Submitted 03/14; published 05/14

# An Efficient Algorithm for Estimating State Sequences in Imprecise Hidden Markov Models

**Jasper De Bock** JASPER.DEBOCK@UGENT.BE
**Gert de Cooman** GERT.DECOOMAN@UGENT.BE
*Ghent University, SYSTeMS Research Group*
*Technologiepark–Zwijnaarde 914*
*9052 Zwijnaarde, Belgium*

## Abstract

We present an efficient exact algorithm for estimating state sequences from outputs or observations in imprecise hidden Markov models (iHMMs). The uncertainty linking one state to the next, and that linking a state to its output, is represented by a set of probability mass functions instead of a single such mass function. We consider as best estimates for state sequences the maximal sequences for the posterior joint state model conditioned on the observed output sequence, associated with a gain function that is the indicator of the state sequence. This corresponds to and generalises finding the state sequence with the highest posterior probability in (precise-probabilistic) HMMs, thereby making our algorithm a generalisation of the one by Viterbi. We argue that the computational complexity of our algorithm is at worst quadratic in the length of the iHMM, cubic in the number of states, and essentially linear in the number of maximal state sequences. An important feature of our imprecise approach is that there may be more than one maximal sequence, typically in those instances where its precise-probabilistic counterpart is sensitive to the choice of prior. For binary iHMMs, we investigate experimentally how the number of maximal state sequences depends on the model parameters. We also present an application in optical character recognition, demonstrating that our algorithm can be usefully applied to robustify the inferences made by its precise-probabilistic counterpart.

## 1. Introduction

In the field of Artificial Intelligence, probabilistic graphical models have become a powerful tool, especially in domains where reasoning under uncertainty is needed (Koller & Friedman, 2009; Pearl, 1988). Usually, this uncertainty is expressed by probabilities, which are estimated from data or elicited from domain experts. However, the assumption that such probabilities can be obtained, or for that matter, that they exist, is not always realistic. This can for example happen when multiple experts disagree, when rounding errors occur, or when the available data is limited; the latter can either be inherent to the problem or a consequence of economic and temporal constraints.

In order to relax this assumption, one can use the theory of *imprecise probabilities*. The basic idea is to allow for sets of probability distributions rather than requiring the specification of a single one. In this way, partial probabilistic information can be expressed easily, for example, by means of linear constraints on probability distributions. This theory of imprecise probability encompasses a number of different, but closely related frameworks; *coherent lower previsions* (Walley, 1991), interval probabilities (Weichselberger, 2000) and belief functions (Dempster, 1967; Shafer, 1976) are well-known examples.

In the context of graphical models, these imprecise-probabilistic ideas have been used to develop the notion of a *credal network* (Cozman, 2000, 2005). It is similar to a *Bayesian network*, but more general in the sense that it allows for local uncertainty models that are imprecisely specified, such as sets of probability distributions. This gain in generality, however, comes at the price of added computational complexity and most existing algorithms are either approximative or cannot handle large networks. In fact, inferences in credal networks are proven to be NP-hard even for singly connected networks with ternary variables (Mauá, de Campos, Benavoli, & Antonucci, 2013).

A notable exception to the intractability of inference problems in credal networks occurs when we drop the so-called strong independence assumption that is usually associated with credal networks and replace it by an assessment of *epistemic irrelevance*. Strong independence requires that the credal network is a convex hull of (precise) Bayesian networks, whereas epistemic irrelevance is a less restrictive property, which is imposed on the imprecise model itself instead of on the individual precise models it consists of; for more information about the difference between these two approaches, see for example the pioneering work of Cozman (2000). Recent work (De Cooman, Hermans, Antonucci, & Zaffalon, 2010) has shown that the use of epistemic irrelevance guarantees that there is an efficient algorithm for updating beliefs about a *single* target node of a credal *tree*, that is essentially linear in the number of nodes in the tree. For imprecise-probabilistic hidden Markov models (iHMMs), which are the credal network equivalent of hidden Markov models (HMMs), this efficiency for single target node inferences has been succesfully exploited to develop an imprecise-probabilistic counterpart to the Kalman filter (Benavoli, Zaffalon, & Miranda, 2011).

In this paper, we tackle the imprecise-probabilistic counterpart of another important application of HMMs: finding the *sequence* of hidden states that has the highest posterior probability conditional on an observed sequence of outputs (Rabiner, 1989). For HMMs with precise local transition and emission probabilities, an efficient dynamic programming algorithm for performing this task was developed by *Viterbi* (1967). For imprecise-probabilistic HMMs however, we know of no algorithm in the literature for which the computational complexity comes even close to that of Viterbi's. We remedy this situation by developing an efficient exact algorithm, called EstiHMM[1], that solves the following imprecise-probabilistic generalisation of the state estimation problem: given an observed sequence of outputs, which are the *maximal* (Troffaes, 2007; Walley, 1991) state sequences for the posterior joint model?

An important difference between our imprecise approach and the more conventional precise-probabilistic approach is that the EstiHMM algorithm may sometimes return more than one solution, whereas the Viterbi algorithm will always produce only a single one. The more imprecise the iHMM is, the more maximal state sequences there will be. For precise HMMs, the EstiHMM and Viterbi algorithms produce identical results. The advantage of this behaviour is that the EstiHMM algorithm will typically return more than one maximal sequence only in those instances where the precise approach is sensitive to the choice of prior. In those cases, the set-valued solution of the EstiHMM algorithm is more likely to contain the correct hidden sequence. Our application in optical character recognition (see Section 9) illustrates this advantage convincingly.

From a credal network point of view, the main contribution of this paper is the EstiHMM algorithm itself. What is especially surprising about this algorithm, is that it provides an efficient solution to an inference problem that deals with multiple target nodes at once, a situation which, in general, is very difficult to handle for current state of the art algorithms in the field. We think that

1. EstiHMM: Estimation in imprecise Hidden Markov Models

the promising results in this paper motivate the study of similar problems for network topologies that go beyond HMMs.

The importance of our results to the HMM community and, by extension, to the field of AI in general, is that they illustrate that model uncertainty—not to be confused with the probabilistic uncertainty that is intrinsic to the model itself—can be dealt with efficiently and can, at the same time, lead to informative, set-valued estimates (sets of maximal state sequences) that can be usefully applied in real-life problems. We believe that model uncertainty is relevant in all subfields of AI where it is difficult—if not impossible—to accurately pinpoint a single probability distribution. Such model uncertainty might have a severe impact on the resulting inferences and, if so, this should be taken into account when basing decisions on these inferences.

We start of in Section 3 by describing imprecise hidden Markov models as a special case of credal trees under epistemic irrelevance. We show in particular how we can use the ideas underlying the MePiCTIr[2] algorithm (De Cooman et al., 2010) to construct a most conservative joint model from imprecise local transition and emission models. We also derive a number of interesting and useful formulas from that construction. The results in this section assume basic knowledge of the theory of coherent lower previsions. We include a short introduction to this theory in Section 2.

In Section 4, we explain the maximality criterion and show how it leads to a set of optimal estimates for the hidden state sequence. Finding all the maximal state sequences seems a daunting task at first: it has a search space that grows exponentially in the length of the Markov chain. However, as shown in Section 5, we can use the basic formulas of Section 3 to derive an appropriate version of Bellman's Principle of Optimality (Bellman, 1957), resulting in an exponential reduction of the search space. By using a number of additional tricks, including a clever reformulation of the maximality criterion, this enables us in Section 6 to devise the EstiHMM algorithm, which efficiently constructs the set of all maximal state sequences.

Section 7 discusses the computational complexity of this EstiHMM algorithm. We show that it is essentially linear in the number of maximal sequences, quadratic in the length of the chain, and cubic in the number of states. We perceive this complexity to be comparable to that of the Viterbi algorithm, especially after realising that the latter makes the simplifying step of resolving ties more or less arbitrarily in order to produce only a single optimal state sequence.

In Section 8, we consider the special case of binary iHMMs, and investigate experimentally how the number of maximal state sequences depends on the model parameters. We comment on the interesting structures that emerge, and provide an heuristic explanation for them. We also demonstrate the algorithm's efficiency by calculating the maximal sequences for an iHMM of length 100.

Finally, in Section 9, we present an application in optical character recognition. It clearly demonstrates the advantages of our algorithm and gives a clear indication that the EstiHMM algorithm is able to robustify the results of the existing Viterbi algorithm in an intelligent manner.

We conclude the paper in Section 10 and discuss a number of possible avenues for future research. In order to make our main argumentation as readable as possible, all technical proofs are relegated to an appendix.

## 2. Freshening Up on Coherent Lower Previsions

We begin with some basic theory of coherent lower previsions; for more information, we refer to Walley's book (1991) and the more recent survey by Miranda (2008).

2. MePiCTIr: Message Passing in Credal Trees under Irrelevance.

Coherent lower previsions are a special type of imprecise probability model. Roughly speaking, whereas classical probability theory assumes that a subject's uncertainty can be represented by a single probability mass function, the theory of imprecise probabilities effectively works with sets of possible probability mass functions, and thereby allows for imprecision as well as indecision to be modelled and represented. For people who are unfamiliar with the theory, looking at it as a way of robustifying the classical theory is perhaps the easiest way to understand and interpret it, and we will use this approach here.

### 2.1 Unconditional Lower Previsions

Let $\mathscr{X}$ be any non-empty, finite[3] set of possible states. We call a real-valued function $f$ on $\mathscr{X}$ a *gamble* and denote the set of all gambles on $\mathscr{X}$ as $\mathscr{G}(\mathscr{X})$. Consider now a set $\mathscr{M}$ of probability mass functions on $\mathscr{X}$. Then with each mass function $p \in \mathscr{M}$, we can associate a *linear prevision*—or expectation functional—$P_p$, defined on $\mathscr{G}(\mathscr{X})$. For every gamble $f \in \mathscr{G}(\mathscr{X})$ , $P_p(f) := \sum_{x\in\mathscr{X}} p(x)f(x)$ is the expected value of $f$, associated with the probability mass function $p$. We now define the *lower prevision*—or lower expectation functional—$\underline{P}_{\mathscr{M}}$ that corresponds with the set $\mathscr{M}$ as the following *lower envelope* of linear previsions:

$$\underline{P}_{\mathscr{M}}(f) := \inf\{P_p(f) \colon p \in \mathscr{M}\} \text{ for all } f \in \mathscr{G}(\mathscr{X}). \tag{1}$$

Similarly, we define the *upper prevision*—or upper expectation functional—$\overline{P}_{\mathscr{M}}$ as

$$\overline{P}_{\mathscr{M}}(f) := \sup\{P_p(f) \colon p \in \mathscr{M}\} = -\inf\{P_p(-f) \colon p \in \mathscr{M}\} = -\underline{P}_{\mathscr{M}}(-f) \text{ for all } f \in \mathscr{G}(\mathscr{X}). \tag{2}$$

We will mostly talk about lower previsions, since it follows from the *conjugacy relation* (2) that the two models are mathematically equivalent.

An *event* $A$ is a subset of the set of possible values $\mathscr{X}$: $A \subseteq \mathscr{X}$. With such an event, we can associate an *indicator* $\mathbb{I}_A$, which is the gamble on $\mathscr{X}$ that assumes the value 1 on $A$, and 0 outside $A$. We call

$$\underline{P}_{\mathscr{M}}(A) := \underline{P}_{\mathscr{M}}(\mathbb{I}_A) = \inf\left\{\sum_{x\in A} p(x) \colon p \in \mathscr{M}\right\}$$

the *lower probability* of the event $A$, and similarly $\overline{P}_{\mathscr{M}}(A) := \overline{P}_{\mathscr{M}}(\mathbb{I}_A)$ its *upper probability*.

It can be shown (Walley, 1991) that the functional $\underline{P}_{\mathscr{M}}$ satisfies the following set of interesting mathematical properties, which define a *coherent lower prevision*:

C1. $\underline{P}_{\mathscr{M}}(f) \geq \min f$ for all $f \in \mathscr{G}(\mathscr{X})$,

C2. $\underline{P}_{\mathscr{M}}(\lambda f) = \lambda \underline{P}_{\mathscr{M}}(f)$ for all $f \in \mathscr{G}(\mathscr{X})$ and real $\lambda \geq 0$, [non-negative homogeneity]

C3. $\underline{P}_{\mathscr{M}}(f+g) \geq \underline{P}_{\mathscr{M}}(f) + \underline{P}_{\mathscr{M}}(g)$ for all $f, g \in \mathscr{G}(\mathscr{X})$. [superadditivity]

Every set of mass functions $\mathscr{M}$ uniquely defines a coherent lower prevision $\underline{P}_{\mathscr{M}}$, but in general the converse does not hold. However, if we limit ourselves to sets of mass functions $\mathscr{M}$ that are closed and convex—which makes them *credal sets*—they are in a one-to-one correspondence with coherent lower previsions (Walley, 1991). This implies that we can use the theory of coherent

3. The theory of coherent lower previsions is applicable to non-finite sets as well, at the expense of some complications. However, for our present purposes, it suffices to consider the finitary case only.

lower previsions as a tool for reasoning with closed convex sets of probability mass functions. From now on, we will no longer explicitly refer to credal sets $\mathscr{M}$, but we will simply talk about coherent lower previsions $\underline{P}$. It is useful to keep in mind that there always is a unique credal set that corresponds with such a coherent lower prevision: $\underline{P} = \underline{P}_{\mathscr{M}}$ for some unique credal set $\mathscr{M}$, given by $\mathscr{M} = \{p\colon (\forall f \in \mathscr{G}(\mathscr{X}))P_p(f) \geq \underline{P}(f)\}$.

A special kind of imprecise model on $\mathscr{X}$ is the *vacuous* lower prevision. It is a model that represents complete ignorance and therefore has the set of all possible mass functions on $\mathscr{X}$ as its credal set $\mathscr{M}$. It can be shown easily that for every $f \in \mathscr{G}(\mathscr{X})$, the corresponding lower prevision is given by $\underline{P}(f) = \min f$.

## 2.2 Conditional Lower Previsions

Conditional lower and upper previsions, which are extensions of the classical conditional expectation functionals, can be defined in a similar, intuitively obvious way: as lower envelopes associated with sets of conditional mass functions.

Consider a variable $X$ in $\mathscr{X}$ and a variable $Y$ in $\mathscr{Y}$. A *conditional lower prevision* $\underline{P}(\cdot|Y)$ on the set $\mathscr{G}(\mathscr{X})$ of all gambles on $\mathscr{X}$ is a two-place real-valued function. For any gamble $f$ on $\mathscr{X}$, $\underline{P}(f|Y)$ is a gamble on $\mathscr{Y}$, whose value $\underline{P}(f|y)$ in $y \in \mathscr{Y}$ is the lower prevision of $f$, *conditional on the event* $Y = y$. If for any $y \in \mathscr{Y}$, the lower prevision $\underline{P}(\cdot|y)$ is coherent—satisfies conditions C1–C3—then we call the conditional lower prevision $\underline{P}(\cdot|Y)$ *separately coherent*. It will sometimes be useful to extend the domain of the conditional lower prevision $\underline{P}(\cdot|y)$ from $\mathscr{G}(\mathscr{X})$ to $\mathscr{G}(\mathscr{X} \times \mathscr{Y})$ by letting $\underline{P}(f|y) := \underline{P}(f(\cdot,y)|y)$ for all gambles $f$ on $\mathscr{X} \times \mathscr{Y}$.

If we have a number of conditional lower previsions involving a number of variables, then each of them must be separately coherent, but we also have to make sure that they satisfy a more stringent *joint coherence* requirement. Explaining this in detail would take us too far; Walley (1991) provides a detailed discussion with motivation. For our present purposes, it suffices to say that joint coherence is very closely related to making sure that these conditional lower previsions are lower envelopes associated with conditional mass functions that satisfy Bayes's Rule.

For a given lower prevision $\underline{P}$ on $\mathscr{G}(\mathscr{X} \times \mathscr{Y})$, there may be more than one corresponding conditional lower prevision $\underline{P}(\cdot|Y)$ that is jointly coherent with $\underline{P}$. Depending on the updating method that is used, one obtains a different model.

If we use *natural extension*, then the conditional coherent lower prevision $\underline{P}(\cdot|Y)$ is defined by $\underline{P}(f|y) := \max\left\{\mu \in \mathbb{R}\colon \underline{P}(\mathbb{I}_{\{y\}}[f-\mu]) \geq 0\right\}$ if $\underline{P}(\{y\}) > 0$ and is vacuous and thus given by $\underline{P}(f|y) := \min f$ if $\underline{P}(\{y\}) = 0$. This is the smallest, most conservative coherent way of conditioning a lower prevision. If $\underline{P}(\{y\}) > 0$, it corresponds to conditioning every probability mass function in the credal set of $\underline{P}$ on the observation that $Y = y$ and taking the lower envelope of all these conditioned mass functions.

If we use *regular extension*, then $\underline{P}(\cdot|Y)$ is defined by $\underline{P}(f|y) := \max\{\mu \in \mathbb{R}\colon \underline{P}(\mathbb{I}_y[f-\mu]) \geq 0\}$ if $\overline{P}(\{y\}) > 0$ and is vacuous if $\overline{P}(\{y\}) = 0$. If $\overline{P}(\{y\}) > 0$, then regular extension (a) gives us the greatest—most informative—conditional lower prevision that is jointly coherent with the original unconditional lower prevision and (b) corresponds to taking all mass functions $p$ in the credal set of $\underline{P}$ for which $p(y) \neq 0$, conditioning them on the observation that $Y = y$ and taking their lower envelope.

Natural and regular extension coincide if $\underline{P}(\{y\}) > 0$ or $\overline{P}(\{y\}) = 0$ but they may differ if $\overline{P}(\{y\}) > \underline{P}(\{y\}) = 0$. In the latter case, natural extension is vacuous, but regular extension usually remains more informative. Furthermore, if $\overline{P}(\{y\}) > 0$, then every coherent updating method yields a conditional lower prevision that lies in between those obtained by natural and regular extension (Walley, 1991; Miranda, 2009).

### 2.3 Different Interpretations for Lower Previsions

As we have just seen, a coherent lower prevision $\underline{P}$ serves as an alternative representation for a closed and convex set $\mathscr{M}$ of probability mass functions. Often, this credal set $\mathscr{M}$ is interpreted as a set of candidates for the one "true" but unknown probability mass function. This interpretation is particularly intuitive for people that are used to working with classical probability theory. Walley (1991, Section 2.10.4) calls this the *sensitivity analysis interpretation*. For the sake of completeness, we mention here that coherent lower previsions can also be given a *behavioural interpretation*, without using the notion of a probability mass function. The lower prevision $\underline{P}(f)$ of a gamble $f \in \mathscr{G}(\mathscr{X})$ is then interpreted as the supremum acceptable buying price that a subject is willing to pay in order to gain the—possibly negative—reward $f(x)$ after the outcome $x \in \mathscr{X}$ of the experiment has been determined. Walley discusses this alternative interpretation extensively.

## 3. Basic Notions

An imprecise hidden Markov model can be depicted using the following probabilistic graphical model:

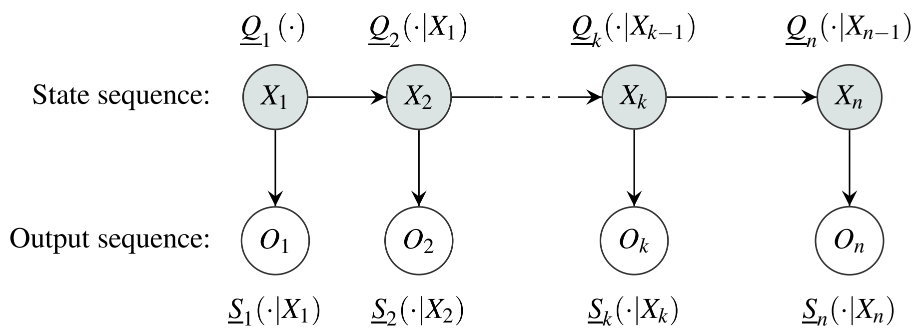


Figure 1: Tree representation of a hidden Markov model

Here $n$ is some natural number. The *state variables* $X_1, \ldots, X_n$ assume values in the respective finite sets $\mathscr{X}_1, \ldots, \mathscr{X}_n$, and the *output variables* $O_1, \ldots, O_n$ assume values in the respective finite sets $\mathscr{O}_1$, $\ldots$, $\mathscr{O}_n$. We denote generic values of $X_k$ by $x_k$, $\hat{x}_k$ or $z_k$, and generic values of $O_k$ by $o_k$.

### 3.1 Local Uncertainty Models

We assume that we have the following local uncertainty models for these variables. For $X_1$, we have a *marginal* lower prevision $\underline{Q}_1$, defined on the set $\mathscr{G}(\mathscr{X}_1)$ of all real-valued maps—or *gambles*—on $\mathscr{X}_1$. For the subsequent states $X_k$, with $k \in \{2, \ldots, n\}$, we have a conditional lower prevision $\underline{Q}_k(\cdot|X_{k-1})$ defined on $\mathscr{G}(\mathscr{X}_k)$, called a *transition model*. In order to maintain uniformity of notation, we will also denote the marginal lower prevision $\underline{Q}_1$ as a conditional lower prevision $\underline{Q}_1(\cdot|X_0)$, where $X_0$ denotes a variable that may only assume a single value $x_0 \in \mathscr{X}_0 := \{x_0\}$, and whose value is therefore certain. For any gamble $f_k$ in $\mathscr{G}(\mathscr{X}_k)$, $\underline{Q}_k(f_k|X_{k-1})$ is interpreted as a gamble on $\mathscr{X}_{k-1}$, whose value $\underline{Q}_k(f_k|x_{k-1})$ in any $x_{k-1} \in \mathscr{X}_{k-1}$ is the lower prevision of the gamble $f_k(X_k)$, conditional on $X_{k-1} = x_{k-1}$.

In addition, for each output $O_k$, with $k \in \{1,\dots,n\}$, we have a conditional lower prevision $\underline{S}_k(\cdot|X_k)$ defined on $\mathscr{G}(\mathscr{O}_k)$, called an *emission model*. For any gamble $g_k$ in $\mathscr{G}(\mathscr{O}_k)$, $\underline{S}_k(g_k|X_k)$ is interpreted as a gamble on $\mathscr{X}_k$, whose value $\underline{S}_k(g_k|x_k)$ in any $x_k \in \mathscr{X}_k$ is the lower prevision of the gamble $g_k(O_k)$, conditional on $X_k = x_k$.

We take all these local—marginal, transition and emission—uncertainty models to be *separately coherent*. Recall that this simply means that for any $k \in \{1,\dots,n\}$, the lower prevision $\underline{Q}_k(\cdot|x_{k-1})$ should be coherent—as an unconditional lower prevision—for all $x_{k-1} \in \mathscr{X}_{k-1}$ and $\underline{S}_k(\cdot|x_k)$ should be coherent for all $x_k \in \mathscr{X}_k$.

### 3.2 Interpretation of the Graphical Structure

We will assume that the graphical representation in Figure 1 represents the following irrelevance assessments: *conditional on its mother variable, the non-parent non-descendants of any variable in the tree are epistemically irrelevant to this variable and its descendants.* We say that a variable $X$ is *epistemically irrelevant* to a variable $Y$ if observing $X$ does not affect our beliefs about $Y$. Mathematically stated in terms of lower previsions: $\underline{P}(f(Y)) = \underline{P}(f(Y)|x)$ for all $f \in \mathscr{G}(\mathscr{Y})$ and all $x \in \mathscr{X}$.

Before we go on, it will be useful to introduce some mathematical short-hand notation for describing joint variables in the tree of Figure 1. For any $1 \leq k \leq \ell \leq n$, we denote the tuple $(X_k, X_{k+1}, \dots, X_\ell)$ by $X_{k:\ell}$, and the tuple $(O_k, O_{k+1}, \dots, O_\ell)$ by $O_{k:\ell}$. $X_{k:\ell}$ is a variable that can assume all values in the set $\mathscr{X}_{k:\ell} := \times_{r=k}^{\ell} \mathscr{X}_r$, and $O_{k:\ell}$ is a variable that can assume all values in the set $\mathscr{O}_{k:\ell} := \times_{r=k}^{\ell} \mathscr{O}_r$. Generic values of $X_{k:\ell}$ are denoted by $x_{k:\ell}$, $\hat{x}_{k:\ell}$ or $z_{k:\ell}$, and generic values of $O_{k:\ell}$ by $o_{k:\ell}$.

**Example 1.** Consider the variable $X_k$ with mother variable $X_{k-1}$ in Figure 1. The variables $X_{1:k-2}$ and $O_{1:k-1}$ are its non-parent non-descendants, and the variables $X_{k+1:n}$ and $O_{k:n}$ its descendants. Our interpretation of the graphical structure of Figure 1 implies that once we know—conditional on—the value $x_{k_1}$ of $X_{k-1}$, additionally learning the values of any of the variables $X_1, \dots, X_{k-2}$ and $O_1, \dots, O_{k-1}$ will not change our beliefs about $X_{k:n}$ and $O_{k:n}$. ♦

### 3.3 Constructing a Global Uncertainty Model

Using the local uncertainty models, we now want to construct a global model: a joint lower prevision $\underline{P}$ on $\mathscr{G}(\mathscr{X}_{1:n} \times \mathscr{O}_{1:n})$ for all the variables $(X_{1:n}, O_{1:n})$ in the tree. This joint lower prevision should (i) be jointly coherent with all the local models; (ii) encode all epistemic irrelevance assessments encoded in the tree; and (iii) be as small, or conservative,[4] as possible. This is a special case of a more general problem for credal trees, discussed and solved in great detail by De Cooman et al. (2010). In this section, we summarise the solution for iHMMs and give an heuristic justification for it; De Cooman et al. prove that the joint model that is presented below is indeed the most conservative lower prevision that is coherent with all the local models and captures all epistemic irrelevance assessments encoded in the tree.

4. Recall that pointwise smaller lower previsions correspond to larger credal sets.

We proceed in a recursive manner. For any $k \in \{1,\dots,n\}$ and any $x_{k-1} \in \mathscr{X}_{k-1}$, we consider the smallest coherent joint lower prevision $\underline{P}_k(\cdot|x_{k-1})$ on $\mathscr{G}(\mathscr{X}_{k:n} \times \mathscr{O}_{k:n})$ for the variables $(X_{k:n}, O_{k:n})$ in the iHMM depicted in Figure 2, representing a subtree of the tree represented in Figure 1, with the lower prevision $\underline{Q}_k(\cdot|x_{k-1})$ acting as the marginal model for the 'first' state variable $X_k$. Note that, due to the notational trick that was introduced in Section 3.1, the global model $\underline{P}$ can be identified with the conditional lower prevision $\underline{P}_1(\cdot|x_0)$.

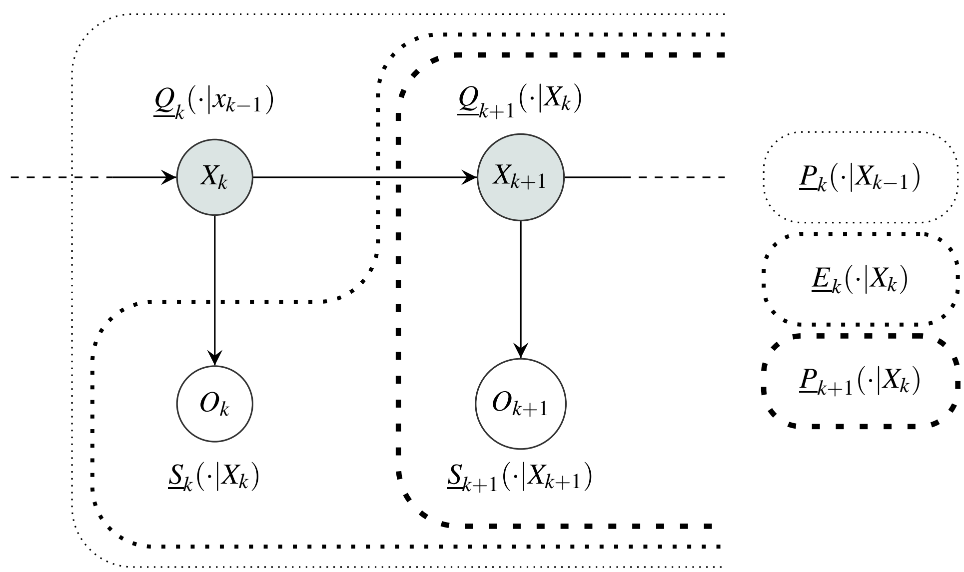


Figure 2: Subtree of the iHMM involving the variables $(X_{k:n}, O_{k:n})$

Our aim is now to develop recursive expressions that enable us to construct $\underline{P}_k(\cdot|x_{k-1})$ out of $\underline{P}_{k+1}(\cdot|X_k)$, $\underline{S}_k(\cdot|X_k)$ and $\underline{Q}_k(\cdot|x_{k-1})$. Using these expressions over and over again will eventually yield the global model $\underline{P} = \underline{P}_1(\cdot|x_0)$.

As a first step, we consider any $x_k \in \mathscr{X}_k$ and combine the joint model $\underline{P}_{k+1}(\cdot|x_k)$ for the variables $(X_{k+1:n}, O_{k+1:n})$, defined on $\mathscr{G}(\mathscr{X}_{k+1:n} \times \mathscr{O}_{k+1:n})$—see the thick dotted lines in Figure 2—,with the local model $\underline{S}_k(\cdot|x_k)$ for the variable $O_k$, defined on $\mathscr{G}(\mathscr{O}_k)$. This will lead to a joint model $\underline{E}_k(\cdot|x_k)$ for the variables $(X_{k+1:n}, O_{k:n})$, defined on $\mathscr{G}(\mathscr{X}_{k+1:n} \times \mathscr{O}_{k:n})$—see the semi-thick dotted lines in Figure 2. This is trivial for $k = n$, since we must have that $\underline{E}_n(\cdot|x_n) = \underline{S}_n(\cdot|x_n)$.

For $k \neq n$, the solution is less obvious. A joint model can be constructed in many different ways, so we will have to impose some conditions. A first condition is that $\underline{E}_k(\cdot|x_k)$ should be a coherent lower prevision that is jointly coherent with the 'marginal' models $\underline{P}_{k+1}(\cdot|x_k)$ and $\underline{S}_k(\cdot|x_k)$. A second, rather obvious, condition is that $\underline{E}_k(\cdot|x_k)$ should coincide with $\underline{P}_{k+1}(\cdot|x_k)$ and $\underline{S}_k(\cdot|x_k)$ on their respective domains. A third condition is that the model should capture the epistemic irrelevance assessments encoded in the tree. In particular these state that, conditional on $X_k = x_k$, the two variables $(X_{k+1:n}, O_{k+1:n})$ and $O_k$ should be *epistemically independent*, or in other words, epistemically irrelevant to one another.

Any model that meets all these conditions is called an *independent product* (De Cooman, Miranda, & Zaffalon, 2011) of $\underline{P}_{k+1}(\cdot|x_k)$ and $\underline{S}_k(\cdot|x_k)$. Generally speaking, such an independent product is not unique. We call the pointwise smallest, most conservative, of all possible independent products, which always exists, the *independent natural extension* (Walley, 1991; De Cooman et al., 2011) of $\underline{P}_{k+1}(\cdot|x_k)$ and $\underline{S}_k(\cdot|x_k)$, and we denote it as $\underline{P}_{k+1}(\cdot|x_k)\otimes\underline{S}_k(\cdot|x_k)$.

Summarising, $\underline{E}_k(\cdot|x_k)$ is given by

$$\underline{E}_k(\cdot|x_k) := \begin{cases} \underline{S}_n(\cdot|x_n) & \text{if } k=n \\ \underline{S}_k(\cdot|x_k)\otimes\underline{P}_{k+1}(\cdot|x_k) & \text{if } k=n-1,\dots,1. \end{cases} \tag{3}$$

The conditionally independent natural extension and its properties were studied in great detail by De Cooman et al. (2011). For the purposes of this paper, it will suffice to recall from that study that—very much like independent products of precise probability models—such independent natural extensions are *factorising*, which implies in particular that

$$\begin{aligned}\underline{E}_k(fg|x_k) = \underline{E}_k(g\underline{E}_k(f|x_k)|x_k) &= \underline{S}_k(g\underline{P}_{k+1}(f|x_k)|x_k)\\ &= \begin{cases} \underline{S}_k(g|x_k)\underline{P}_{k+1}(f|x_k) & \text{if } \underline{P}_{k+1}(f|x_k)\geq 0 \\ \overline{S}_k(g|x_k)\underline{P}_{k+1}(f|x_k) & \text{if } \underline{P}_{k+1}(f|x_k)\leq 0 \end{cases}\\ &= \overline{\underline{S}}_k(g|x_k)\odot\underline{P}_{k+1}(f|x_k),\end{aligned} \tag{4}$$

for all $f\in\mathscr{G}(\mathscr{X}_{k+1:n}\times\mathscr{O}_{k+1:n})$ and all *non-negative* $g\in\mathscr{G}(\mathscr{O}_k)$—we call a gamble non-negative if all its values are. In this expression, the first equality is the actual factorisation property. The second equality holds because $\underline{E}_k(\cdot|x_k)$ coincides with $\underline{P}_{k+1}(\cdot|x_k)$ and $\underline{S}_k(\cdot|x_k)$ on their respective domains. The third equality follows from the conjugacy relation—Equation (2)—and coherence condition C2, and for the fourth we have used the shorthand notation $\overline{\underline{m}}\odot x := \underline{m}\max\{0,x\}+\overline{m}\min\{0,x\}$. Further on, we will also use the analogous notation $\overline{\underline{m}}\,\overline{\underline{n}}\odot x := \underline{m}\,\underline{n}\max\{0,x\}+\overline{m}\,\overline{n}\min\{0,x\}$.

In a second and final step, we combine the joint model $\underline{E}_k(\cdot|X_k)$ for the variables $(X_{k+1:n},O_{k:n})$, defined on $\mathscr{G}(\mathscr{X}_{k+1:n}\times\mathscr{O}_{k:n})$, with the local model $\underline{Q}_k(\cdot|x_{k-1})$ for the variable $X_k$, defined on $\mathscr{G}(\mathscr{X}_k)$, into the joint model $\underline{P}_k(\cdot|x_{k-1})$ for the variables $(X_{k:n},O_{k:n})$, defined on $\mathscr{G}(\mathscr{X}_{k:n}\times\mathscr{O}_{k:n})$. It has been shown elsewhere (Miranda & de Cooman, 2007; Walley, 1991) that the most conservative coherent way of doing this, is by means of *marginal extension*, also known as the law of iterated lower expectations. This leads to $\underline{P}_k(\cdot|x_{k-1}) := \underline{Q}_k(\underline{E}_k(\cdot|X_k)|x_{k-1})$, or, if we now allow $x_{k-1}$ to range over $\mathscr{X}_{k-1}$:

$$\underline{P}_k(\cdot|X_{k-1}) := \underline{Q}_k(\underline{E}_k(\cdot|X_k)|X_{k-1}). \tag{5}$$

For practical purposes, it is useful to see that this is equivalent with

$$\underline{P}_k(f|X_{k-1}) = \underline{Q}_k\left(\sum_{x_k\in\mathscr{X}_k}\mathbb{I}_{\{x_k\}}\underline{E}_k(f(x_k,X_{k+1:n},O_{k:n})|x_k)\middle|X_{k-1}\right)$$

for all $f\in\mathscr{G}(\mathscr{X}_{k:n}\times\mathscr{O}_{k:n})$. Recall that in this expression, the *indicator* $\mathbb{I}_{\{x_k\}}$ is a gamble on $\mathscr{X}_k$ that assumes the value 1 if $X_k=x_k$ and 0 if $X_k\neq x_k$.

### 3.4 Interesting Lower and Upper Probabilities

Without too much trouble,[5] we can use Equations (3)–(5) to derive the following expressions for a number of interesting lower and upper probabilities:

$$\underline{P}_k(\{o_{k:n}\} \times \{x_{k:n}\}|x_{k-1}) = \prod_{i=k}^{n} \underline{S}_i(\{o_i\}|x_i)\underline{Q}_i(\{x_i\}|x_{i-1}) \tag{6}$$

$$\overline{P}_k(\{o_{k:n}\} \times \{x_{k:n}\}|x_{k-1}) = \prod_{i=k}^{n} \overline{S}_i(\{o_i\}|x_i)\overline{Q}_i(\{x_i\}|x_{i-1}) \tag{7}$$

for all $x_{k-1} \in \mathscr{X}_{k-1}$, $x_{k:n} \in \mathscr{X}_{k:n}$, $o_{k:n} \in \mathscr{O}_{k:n}$ and $k \in \{1,\dots,n\}$, and

$$\underline{E}_k(\{o_{k:n}\} \times \{x_{k+1:n}\}|x_k) = \underline{S}_k(\{o_k\}|x_k) \prod_{i=k+1}^{n} \underline{S}_i(\{o_i\}|x_i)\underline{Q}_i(\{x_i\}|x_{i-1}) \tag{8}$$

$$\overline{E}_k(\{o_{k:n}\} \times \{x_{k+1:n}\}|x_k) = \overline{S}_k(\{o_k\}|x_k) \prod_{i=k+1}^{n} \overline{S}_i(\{o_i\}|x_i)\overline{Q}_i(\{x_i\}|x_{i-1}). \tag{9}$$

for all $x_k \in \mathscr{X}_k$, $x_{k+1:n} \in \mathscr{X}_{k+1:n}$, $o_{k:n} \in \mathscr{O}_{k:n}$ and $k \in \{1,\dots,n\}$. Recall that we equate events with their indicators, and that the lower and upper prevision of these indicators correspond to the lower and upper probability of that event; see Section 2. For example, in Equation (6), $\underline{P}_k(\{o_{k:n}\} \times \{x_{k:n}\}|x_{k-1}) := \underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{x_{k:n}\}}|x_{k-1})$ is the lower probability that, conditional on $X_{k-1} = x_{k-1}$, the rest of the hidden sequence has the value $x_{k:n}$, with corresponding observations $o_{k:n}$. This joint lower probablity is obtained simply by multiplying the relevant local lower (transition and emission) probabilities.

We will assume throughout that

$$\overline{P}(\{x_{1:n}\} \times \{o_{1:n}\}) > 0 \text{ for all } x_{1:n} \in \mathscr{X}_{1:n} \text{ and } o_{1:n} \in \mathscr{O}_{1:n}$$

or, equivalently—by Equation (7), for $k = 1$—, that all *local upper probabilities are positive*, in the sense that (De Cooman et al., 2010):

$$\overline{Q}_k(\{x_k\}|x_{k-1}) > 0 \text{ and } \overline{S}_k(\{o_k\}|x_k) > 0 \quad \text{for all } k \in \{1,\dots,n\},\ x_{k-1} \in \mathscr{X}_{k-1},\ x_k \in \mathscr{X}_k \text{ and } o_k \in \mathscr{O}_k. \tag{10}$$

This assumption is very weak and not at all restrictive for practical purposes. The imprecise-probabilistic local models are often constructed by adding some margin of error around a precise model, thereby making all upper transition probabilities positive by construction. We will however allow lower transition probabilities to be zero, which is something that does happen often in practical problems.

**Proposition 1.** *If all local upper probabilities are positive—Equation* (10)*—, then we have for all* $k \in \{1,\dots,n\}$, $x_k \in \mathscr{X}_k$, $x_{k-1} \in \mathscr{X}_{k-1}$ *and* $o_{k:n} \in \mathscr{O}_{k:n}$ *that* $\overline{P}_k(\{o_{k:n}\}|x_{k-1}) > 0$ *and* $\overline{E}_k(\{o_{k:n}\}|x_k) > 0$.

5. As an example, we derive Equations (6) and (7) in Appendix A.

## 4. Estimating States from Outputs

In a hidden Markov model, the states are not directly observable, but the outputs are, and the general aim is to use the outputs to estimate the states. We concentrate on the following problem: *Suppose we have observed the output sequence $o_{1:n}$, estimate the state sequence $x_{1:n}$*. We will use an essentially Bayesian approach to do so, but need to allow for the fact that we are working with imprecise rather than precise probability models. We consider as optimal estimates all state sequences that are maximal, a criterion which we introduce in Section 4.2; see Section 4.3 for two alternative criteria, which we will not consider further in the context of this paper. The main contribution of this section is a formulation of maximality that is stated directly in terms of the unconditional global model $\underline{P}$, instead of the conditional model $\underline{P}(\cdot|o_{1:n})$ that is conventionally used for this purpose. Furthermore, and rather surprisingly, this alternative formulation is valid regardless of whether we use regular or natural extension to derive $\underline{P}(\cdot|o_{1:n})$ from $\underline{P}$.

### 4.1 Updating the iHMM

The first step in our approach consists in updating (or conditioning) the joint model $\underline{P} := \underline{P}_1(\cdot|x_0)$ on the observed outputs $O_{1:n} = o_{1:n}$. As mentioned in Section 2, there is no unique coherent way to perform this updating. However, for the particular problem we are solving in this paper, it so happens that it makes no difference which updating method is used, as long as it is coherent. For the time being, we use *regular extension*, but later on in Section 4.2, we will show that any other coherent updating method yields the same results.

Since it follows from the positivity assumption (10) and Proposition 1 that $\overline{P}(\{o_{1:n}\}) > 0$, regular extension leads us to consider the updated lower prevision $\underline{P}(\cdot|o_{1:n})$ on $\mathscr{G}(\mathscr{X}_{1:n})$, given by:

$$\underline{P}(f|o_{1:n}) := \max\left\{\mu \in \mathbb{R} \colon \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[f-\mu]) \geq 0\right\} \text{ for all gambles } f \text{ on } \mathscr{X}_{1:n}. \tag{11}$$

Using the coherence of the joint lower prevision $\underline{P}$, it is not hard to prove that when $\underline{P}(\{o_{1:n}\}) > 0$, $\underline{P}(\mathbb{I}_{\{o_{1:n}\}}[f-\mu])$ is a strictly decreasing and continuous function of $\mu$, which therefore has a unique zero—see Lemma 7(i)&(iii) in Appendix A. As a consequence, we have for any $f \in \mathscr{G}(\mathscr{X}_{1:n})$ that

$$\underline{P}(f|o_{1:n}) \leq 0 \Leftrightarrow (\forall \mu > 0)\underline{P}(\mathbb{I}_{\{o_{1:n}\}}[f-\mu]) < 0 \Leftrightarrow \underline{P}(\mathbb{I}_{\{o_{1:n}\}}f) \leq 0. \tag{12}$$

In fact, it is not hard to infer from the strictly decreasing and continuous character of $\underline{P}(\mathbb{I}_{\{o_{1:n}\}}[f-\mu])$ that $\underline{P}(f|o_{1:n})$ and $\underline{P}(\mathbb{I}_{\{o_{1:n}\}}f)$ have the same sign. They are either both negative, both positive or both equal to zero; see also Figure 3.

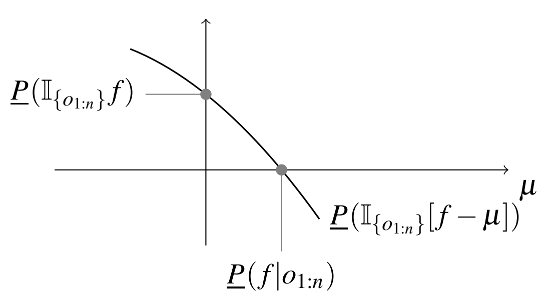


Figure 3: Conditional versus unconditional lower prevision

Equation (12) will be of crucial importance further on. However, in general, we want to allow $\underline{P}(\{o_{1:n}\})$ to be zero—since this may happen if you allow lower transition probabilities to be zero—, while requiring that $\overline{P}(\{o_{1:n}\}) > 0$—because this follows from the positivity assumption (10) and Proposition 1. This will, generally speaking, invalidate the second equivalence in Equation (12): it turns into an implication only. But, if we limit ourselves to the specific type of gambles on $\mathscr{X}_{1:n}$ of the form $f = \mathbb{I}_{\{\hat{x}_{1:n}\}} - \mathbb{I}_{\{x_{1:n}\}}$, we can still prove the following important theorem.

**Theorem 2.** *If all local upper probabilities are positive—Equation* (10)—*, then for fixed values of* $x_{1:n}, \hat{x}_{1:n} \in \mathscr{X}_{1:n}$ *and* $o_{1:n} \in \mathscr{O}_{1:n}$*, we have that* $\underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}])$ *and* $\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n})$ *have the same sign. They are both positive, both negative or both zero.*

### 4.2 Maximal State Sequences

The next step now consists in using the posterior model $\underline{P}(\cdot|o_{1:n})$ to find best estimates for the state sequence $x_{1:n}$. On the Bayesian approach, this is usually done by solving a decision-making, or optimisation problem: we associate a gain function $\mathbb{I}_{\{x_{1:n}\}}$ with every candidate state sequence $x_{1:n}$, and select as best estimates those state sequences $\hat{x}_{1:n}$ that maximise the posterior expected gain, resulting in state sequences with maximal posterior probability.

Here we generalise this decision-making approach towards working with imprecise probability models. The criterion we use to decide which estimates are optimal for the given gain functions is that of (Walley–Sen) *maximality* (Troffaes, 2007; Walley, 1991). Maximality has a number of very desirable properties that make sure it works well in optimisation contexts (De Cooman & Troffaes, 2005; Huntley & Troffaes, 2010), and it is well-justified from a behavioural point of view, as well as in a robustness approach, as we shall see presently.

We can express a strict preference $\succ$ between two state sequence estimates $\hat{x}_{1:n}$ and $x_{1:n}$ as follows:

$$\hat{x}_{1:n} \succ x_{1:n} \Leftrightarrow \underline{P}(\mathbb{I}_{\{\hat{x}_{1:n}\}} - \mathbb{I}_{\{x_{1:n}\}}|o_{1:n}) > 0.$$

On a behavioural interpretation, this expresses that a subject with lower prevision $\underline{P}(\cdot|o_{1:n})$ is disposed to pay some strictly positive amount of utility to replace the gain associated with the estimate $x_{1:n}$ with the gain associated with the estimate $\hat{x}_{1:n}$; Walley (1991, Section 3.9) provides additional information. Alternatively, from a robustness point of view, this expresses that for each conditional mass function $p(\cdot|o_{1:n})$ in the credal set associated with the updated lower prevision $\underline{P}(\cdot|o_{1:n})$, the state sequence $\hat{x}_{1:n}$ has a posterior probability $p(\hat{x}_{1:n}|o_{1:n})$ that is *strictly higher* than the posterior probability $p(x_{1:n}|o_{1:n})$ of the state sequence $x_{1:n}$.

The binary relation $\succ$ thus defined is a strict partial order—an irreflexive and transitive binary relation—on the set of state sequences $\mathscr{X}_{1:n}$, and we consider an estimate $\hat{x}_{1:n}$ to be *optimal* when it is *undominated*, or *maximal*, in this strict partial order:

$$\begin{aligned}\hat{x}_{1:n} \in \operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n}) &\Leftrightarrow (\forall x_{1:n} \in \mathscr{X}_{1:n}) x_{1:n} \not\succ \hat{x}_{1:n} \\ &\Leftrightarrow (\forall x_{1:n} \in \mathscr{X}_{1:n}) \underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n}) \leq 0 \\ &\Leftrightarrow (\forall x_{1:n} \in \mathscr{X}_{1:n}) \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}]) \leq 0, \end{aligned} \tag{13}$$

where the very useful last equivalence follows from Theorem 2. *In summary then, the aim of this paper is to develop an efficient algorithm for finding the set of maximal estimates* $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$.

Our statement in Section 4.1, that any coherent updating method would yield the same results as regular extension, can now be justified. Since coherent updating is unique if $\underline{P}(\{o_{1:n}\}) > 0$, and since the case $\overline{P}(\{o_{1:n}\}) = 0$ is excluded by Proposition 1 and our positivity assumption (10), we only need to motivate our statement in the special case that $\underline{P}(\{o_{1:n}\}) = 0$ and $\overline{P}(\{o_{1:n}\}) > 0$.

If we use regular extension to update our model, then the optimal estimates are given by Equation (13). For the special case $\underline{P}(\{o_{1:n}\}) = 0$, we find for all $x_{1:n} \in \mathscr{X}_{1:n}$ and $\hat{x}_{1:n} \in \mathscr{X}_{1:n}$ that

$$\underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}]) \leq \underline{P}(\mathbb{I}_{\{o_{1:n}\}}) = \underline{P}(\{o_{1:n}\}) = 0,$$

where the first inequality follows from the monotonicity of coherent lower previsions (as a consequence of C1 and C2). Therefore, we find that if $\underline{P}(\{o_{1:n}\}) = 0$, then all sequences are optimal, resulting in $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n}) = \mathscr{X}_{1:n}$.

If we use natural extension to update our joint model, then the optimal state sequences are still given by Equation (13), but the final equivalence no longer holds because it uses Theorem 2, which assumes the use of regular extension to perform updating of the joint model. However, for the special case of $\underline{P}(\{o_{1:n}\}) = 0$, natural extension by definition leads to the updated model being equal to the vacuous one. Therefore, we find for all $x_{1:n} \in \mathscr{X}_{1:n}$ and $\hat{x}_{1:n} \in \mathscr{X}_{1:n}$ that

$$\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n}) = \min(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}) \leq 0.$$

This implies that for the special case where $\underline{P}(\{o_{1:n}\}) = 0$ and $\overline{P}(\{o_{1:n}\}) > 0$—identical to what we found for regular extension—natural extension also results in all sequences being optimal, meaning that $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n}) = \mathscr{X}_{1:n}$.

We have thus shown that, even in the special case that $\underline{P}(\{o_{1:n}\}) = 0$ and $\overline{P}(\{o_{1:n}\}) > 0$, the set of optimal sequences is the same, regardless of whether we use natural or regular extension to update our joint model. Since in that special case, every other coherent updating method lies in between these two methods, all of them are bound to yield the same $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$. We can therefore conclude that the results in this paper do not depend on the particular updating method that is chosen, as long as it is coherent.

### 4.3 Other Decision Criteria

Instead of looking for maximal state sequences, one could use other decision criteria as well (Troffaes, 2007), two of which we discuss in the present section.

A first approach that we will not consider further on, consists in trying to find the so-called $\Gamma$-*maximin* state sequences $\overline{x}_{1:n}$, which maximise the posterior lower probability:

$$\overline{x}_{1:n} \in \operatorname*{argmax}_{x_{1:n} \in \mathscr{X}_{1:n}} \underline{P}(\{x_{1:n}\}|o_{1:n}).$$

This approach basically optimises the worst-case scenario—the lower probability—and can therefore be regarded as a risk averse choice. From a computational point of view, finding these $\Gamma$-maximin sequences is a rather complicated affair. Not only do we need to optimise over an exponential number of sequences, but on top of that, every single lower probability $\underline{P}(\{x_{1:n}\}|o_{1:n})$ in this optimisation problem is hard to compute. On the positive side, we have recently discovered that—in the case of epistemic irrelevance—it is possible to calculate these lower probabilities efficiently in a recursive manner. However, these results are not published yet and fall beyond the scope of the current paper. We know of no other algorithm that can calculate these lower probabilities efficiently. In any case, the issue still remains that we need to optimise over the exponentially large set $\mathscr{X}_{1:n}$.

A second approach that will not be considered further on consists in working with the so-called *E-admissible* sequences, which are those sequences that maximise the expected gain for at least one conditional mass function $p(\cdot|o_{1:n})$ in the credal set associated with the updated lower prevision $\underline{P}(\cdot|o_{1:n})$. If one interprets an imprecise model as a collection—a credal set—of precise models, one of which is the unknown "true" model, then one of these E-admissible solutions is the unknown "true" solution. E-admissible state sequences are very difficult to compute. The intuitive reason is that we need to solve the "precise problem" for every $p(\cdot|o_{1:n})$ in the credal set associated with $\underline{P}(\cdot|o_{1:n})$, of which there are infinitely many. State of the art algorithms (Kikuti, Cozman, & de Campos, 2005; Utkin & Augustin, 2005) avoid this issue, but are still quadratic in the search space. This makes them intractable for the present problem because our search space $\mathscr{X}_{1:n}$ is exponential in the length of the iHMM.

Besides the computational difficulties with the other approaches, there are a number of additional reasons why, in this paper, we focus on maximal state sequences rather than Γ-maximin or E-admissible ones. The first and most important reason is that we were able to develop an algorithm that can determine them efficiently; see Sections 6 and 7. Secondly, and this is a common advantage of maximality and E-admissibility: the higher the imprecision of the model, the more solutions are returned. In contrast, even for high imprecision, in most cases, there will be only one Γ-maximin sequence (except if two or more sequences have the same highest conditional lower probability). Our application in Section 9 clearly illustrates that emitting more than a single solution can indeed be useful. Thirdly, in those cases where other decision criteria are preferred, maximal state sequences can still be of use because every Γ-maximin and E-admissible state sequence is guaranteed to be maximal as well (Troffaes, 2007). If our algorithm yields only a single maximal solution, this is also the unique Γ-maximin and E-admissible solution. If more than one maximal sequence is returned, this can be regarded as preprocessing step. For example, once we know all maximal solutions, finding the Γ-maximin solutions amounts to comparing the posterior lower probabilities of these maximal sequences only, instead of all sequences in $\mathscr{X}_{1:n}$.

### 4.4 Maximal Subsequences

We shall see below that in order to find the set of maximal estimates, it is useful to consider more general sets of so-called maximal subsequences: for any $k \in \{1,\ldots,n\}$ and $x_{k-1} \in \mathscr{X}_{k-1}$, we define $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$:

$$\hat{x}_{k:n} \in \operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) \Leftrightarrow (\forall x_{k:n} \in \mathscr{X}_{k:n})\ \underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) \leq 0. \tag{14}$$

The interpretation of these sets is immediate if we consider the part of the original iHMM that is depicted in Figure 4, where we take $\underline{Q}_k(\cdot|x_{k-1})$ as the marginal model for the first state $X_k$. Then, as explained in Section 3.3, the corresponding joint lower prevision on $\mathscr{G}(\mathscr{X}_{k:n} \times \mathscr{O}_{k:n})$ is precisely $\underline{P}_k(\cdot|x_{k-1})$, and if we have a sequence of outputs $o_{k:n}$, then $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$ is the set of state sequence estimates that are undominated by any other estimate in $\mathscr{X}_{k:n}$. It should be clear that the set $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$ we are eventually looking for, can also be written as $\operatorname{opt}(\mathscr{X}_{1:n}|x_0,o_{1:n})$.

### 4.5 Useful Recursion Equations

Fix any $k$ in $\{1,\ldots,n\}$. If we look at Equation (14), we see that it will be useful to derive a manageable expression for the lower prevision $\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1})$. This can be easily done—see

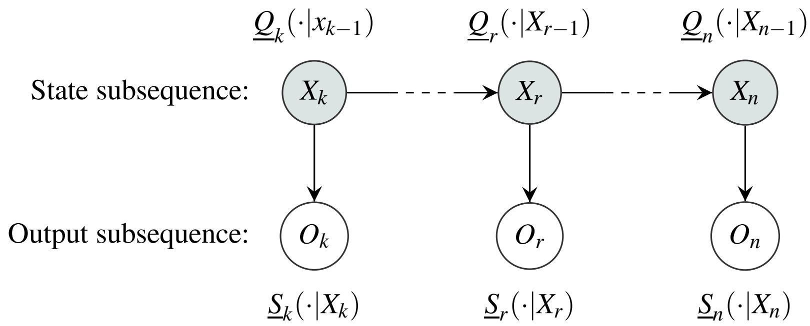


Figure 4: Tree representation of a part of the original iHMM

Appendix A—by using Equations (3)–(7) and a few algebraic manipulations. We consider three different cases. If $\hat{x}_k = x_k$ and $k \in \{1, \dots, n-1\}$ then, using the notation introduced in Section 3.3:

$$\begin{aligned}\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) \\ = \overline{Q}_k(\{\hat{x}_k\}|x_{k-1})\overline{S}_k(\{o_k\}|\hat{x}_k) \odot \underline{P}_{k+1}(\mathbb{I}_{\{o_{k+1:n}\}}[\mathbb{I}_{\{x_{k+1:n}\}} - \mathbb{I}_{\{\hat{x}_{k+1:n}\}}]|\hat{x}_k).\end{aligned} \tag{15}$$

If $\hat{x}_n = x_n$ then

$$\underline{P}_n(\mathbb{I}_{\{o_n\}}[\mathbb{I}_{\{x_n\}} - \mathbb{I}_{\{\hat{x}_n\}}]|x_{n-1}) = 0. \tag{16}$$

If $\hat{x}_k \neq x_k$ and $k \in \{1, \dots, n\}$ then

$$\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) = \underline{Q}_k(\mathbb{I}_{\{x_k\}}\beta(x_{k:n}) - \mathbb{I}_{\{\hat{x}_k\}}\alpha(\hat{x}_{k:n})|x_{k-1}), \tag{17}$$

where we define, for any $x_{k:n} \in \mathscr{X}_{k:n}$:

$$\beta(x_{k:n}) \coloneqq \underline{E}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{x_{k+1:n}\}}|x_k) = \underline{S}_k(\{o_k\}|x_k)\prod_{i=k+1}^{n}\underline{S}_i(\{o_i\}|x_i)\underline{Q}_i(\{x_i\}|x_{i-1}) \tag{18}$$

$$\alpha(x_{k:n}) \coloneqq \overline{E}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{x_{k+1:n}\}}|x_k) = \overline{S}_k(\{o_k\}|x_k)\prod_{i=k+1}^{n}\overline{S}_i(\{o_i\}|x_i)\overline{Q}_i(\{x_i\}|x_{i-1}). \tag{19}$$

It is useful to realise that $\beta(x_{k:n})$ and $\alpha(x_{k:n})$ are just shorthand notations for the lower and upper probabilities in Equations (8) and (9), for a fixed sequence of observations. For any given sequence of states $x_{k:n} \in \mathscr{X}_{k:n}$, the $\alpha(x_{k:n})$ and $\beta(x_{k:n})$ can be found by simple backward recursion:

$$\alpha(x_{k:n}) \coloneqq \alpha(x_{k+1:n})\overline{S}_k(\{o_k\}|x_k)\overline{Q}_{k+1}(\{x_{k+1}\}|x_k) \tag{20}$$

$$\beta(x_{k:n}) \coloneqq \beta(x_{k+1:n})\underline{S}_k(\{o_k\}|x_k)\underline{Q}_{k+1}(\{x_{k+1}\}|x_k), \tag{21}$$

for $k \in \{1, \dots, n-1\}$, and starting from:

$$\alpha(x_{n:n}) = \alpha(x_n) \coloneqq \overline{S}_n(\{o_n\}|x_n) \text{ and } \beta(x_{n:n}) = \beta(x_n) \coloneqq \underline{S}_n(\{o_n\}|x_n). \tag{22}$$

## 5. The Principle of Optimality

Determining the state sequences in $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$ directly using Equation (13) clearly has a complexity that is exponential in the length of the chain. We are now going to take a dynamic programming approach (Bellman, 1957) to reducing this complexity by deriving a recursion equation for the sets of optimal (sub)sequences $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$.

**Theorem 3** (Principle of Optimality)**.** *For $k \in \{1,\ldots,n-1\}$, all $x_{k-1} \in \mathscr{X}_{k-1}$ and all $\hat{x}_{k:n} \in \mathscr{X}_{k:n}$: if $\underline{Q}_k(\{\hat{x}_k\}|x_{k-1}) > 0$ and $\underline{S}_k(\{o_k\}|\hat{x}_k) > 0$, then*

$$\hat{x}_{k:n} \in \operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) \Rightarrow \hat{x}_{k+1:n} \in \operatorname{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k,o_{k+1:n}).$$

As an immediate consequence, we find that

$$\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) \subseteq \operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}), \tag{23}$$

where the set $\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$ consists of all the sequences in $\mathscr{X}_{k:n}$ that can still be an element of $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$ according to Theorem 3:

$$\begin{aligned}\operatorname{cand}&(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})\\ &:= \Bigg(\bigcup_{x_k \in \operatorname{Pos}_k(x_{k-1})} x_k \oplus \operatorname{opt}(\mathscr{X}_{k+1:n}|x_k,o_{k+1:n})\Bigg) \cup \Bigg(\bigcup_{x_k \notin \operatorname{Pos}_k(x_{k-1})} x_k \oplus \mathscr{X}_{k+1:n}\Bigg).\end{aligned} \tag{24}$$

Here $\oplus$ denotes concatenation of state sequences and the set of states $\operatorname{Pos}_k(x_{k-1}) \subseteq \mathscr{X}_k$ is defined as

$$x_k \in \operatorname{Pos}_k(x_{k-1}) \Leftrightarrow \underline{Q}_k(\{x_k\}|x_{k-1}) > 0 \text{ and } \underline{S}_k(\{o_k\}|x_k) > 0. \tag{25}$$

Equation (24) simplifies to

$$\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) = \bigcup_{x_k \in \mathscr{X}_k} x_k \oplus \operatorname{opt}(\mathscr{X}_{k+1:n}|x_k,o_{k+1:n}) \tag{26}$$

if all local lower probabilities are positive, but this is not generally true in the more general case we are considering here, where only the upper probabilities are required to be positive.

## 6. An Algorithm for Finding all Maximal State Sequences

We now use Equation (23) to devise an algorithm for constructing the set $\operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$ of maximal state sequences in a recursive manner.

### 6.1 Initial Set-up Using Backward Recursion

We begin by defining a few auxiliary notions. First of all, we consider the following thresholds:

$$\theta_k(\hat{x}_k,x_k|x_{k-1}) := \min\Big\{a \geq 0\colon \underline{Q}_k(\mathbb{I}_{\{x_k\}} - a\mathbb{I}_{\{\hat{x}_k\}}|x_{k-1}) \leq 0\Big\} \tag{27}$$

for all $k \in \{1,\ldots,n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$ and $x_1,\hat{x}_1 \in \mathscr{X}_1$ such that $x_1 \neq \hat{x}_1$.

Next, we define

$$\alpha_k^{\max}(x_k) := \max_{\substack{z_{k:n}\in\mathscr{X}_{k:n}\\ z_k=x_k}} \alpha(z_{k:n}) \text{ and } \beta_k^{\max}(x_k) := \max_{\substack{z_{k:n}\in\mathscr{X}_{k:n}\\ z_k=x_k}} \beta(z_{k:n}) \tag{28}$$

for all $k \in \{1,\dots,n\}$ and $x_k \in \mathscr{X}_k$. Using Equations (20)–(21), these can be calculated efficiently using the following backward recursive (dynamic programming) procedure:

$$\begin{aligned}\alpha_k^{\max}(x_k) &= \max_{x_{k+1}\in\mathscr{X}_{k+1}} \alpha_{k+1}^{\max}(x_{k+1})\overline{S}_k(\{o_k\}|x_k)\overline{Q}_{k+1}(\{x_{k+1}\}|x_k)\\ &= \overline{S}_k(\{o_k\}|x_k) \max_{x_{k+1}\in\mathscr{X}_{k+1}} \alpha_{k+1}^{\max}(x_{k+1})\overline{Q}_{k+1}(\{x_{k+1}\}|x_k),\end{aligned} \tag{29}$$

and

$$\begin{aligned}\beta_k^{\max}(x_k) &= \max_{x_{k+1}\in\mathscr{X}_{k+1}} \beta_{k+1}^{\max}(x_{k+1})\underline{S}_k(\{o_k\}|x_k)\underline{Q}_{k+1}(\{x_{k+1}\}|x_k)\\ &= \underline{S}_k(\{o_k\}|x_k) \max_{x_{k+1}\in\mathscr{X}_{k+1}} \beta_{k+1}^{\max}(x_{k+1})\underline{Q}_{k+1}(\{x_{k+1}\}|x_k),\end{aligned} \tag{30}$$

for $k \in \{1,\dots,n-1\}$, starting from

$$\alpha_n^{\max}(x_n) = \alpha(x_n) = \overline{S}_n(\{o_n\}|x_n) \text{ and } \beta_n^{\max}(x_n) = \beta(x_n) = \underline{S}_n(\{o_n\}|x_n). \tag{31}$$

Finally, we let

$$\alpha_k^{\text{opt}}(\hat{x}_k|x_{k-1}) := \max_{\substack{x_k\in\mathscr{X}_k\\ x_k\neq\hat{x}_k}} \beta_k^{\max}(x_k)\theta_k(\hat{x}_k,x_k|x_{k-1}), \tag{32}$$

for all $k \in \{1,\dots,n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$ and $\hat{x}_k \in \mathscr{X}_k$.

### 6.2 A Recursive Solution Method

It turns out that the $\alpha_k^{\text{opt}}(\hat{x}_k|x_{k-1})$, calculated by Equation (32), are extremely useful. As proved in Appendix A, they allow us to significantly simplify Equation (14) as follows:

$$\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) = \left\{\hat{x}_{k:n} \in \operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n}) \colon \alpha(\hat{x}_{k:n}) \geq \alpha_k^{\text{opt}}(\hat{x}_k|x_{k-1})\right\}, \tag{33}$$

which, for $k=n$, reduces to

$$\operatorname{opt}(\mathscr{X}_n|x_{n-1},o_n) = \left\{\hat{x}_n \in \mathscr{X}_n \colon \alpha(\hat{x}_n) \geq \alpha_n^{\text{opt}}(\hat{x}_n|x_{n-1})\right\}. \tag{34}$$

Since $\operatorname{opt}(\mathscr{X}_{1:n}|x_0,o_{1:n}) = \operatorname{opt}(\mathscr{X}_{1:n}|o_{1:n})$, this suggest the following algorithm for constructing the set of all maximal state sequences.

**Algorithm 1:** ConstructMaximals

**Data**: the local lower and upper probabilities and the parameters $\alpha_k^{\max}$ and $\alpha_k^{\mathrm{opt}}$ (calculated as in Section 6.1)
**Result**: the set of all maximal state sequences: $\mathrm{opt}\left(\mathscr{X}_{1:n}|o_{1:n}\right)$

1 **for** $x_{n-1} \in \mathscr{X}_{n-1}$ **do**
2     $\mathrm{opt}\left(\mathscr{X}_n|x_{n-1}, o_n\right) \leftarrow \emptyset$
3     **for** $\hat{x}_n \in \mathscr{X}_n$ **do**
4         **if** $\alpha(\hat{x}_n) \geq \alpha_n^{\mathrm{opt}}(\hat{x}_n|x_{n-1})$ **then** add $\hat{x}_n$ to $\mathrm{opt}\left(\mathscr{X}_n|x_{n-1}, o_n\right)$
5 **for** $k \leftarrow n-1$ **to** $1$ **do**
6     **for** $x_{k-1} \in \mathscr{X}_{k-1}$ **do**
7         $\mathrm{opt}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right) \leftarrow \emptyset$
8         **for** $\hat{x}_{k:n} \in \mathrm{cand}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right)$ **do**
9             **if** $\alpha(\hat{x}_{k:n}) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$ **then** add $\hat{x}_{k:n}$ to $\mathrm{opt}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right)$
10 **return** $\mathrm{opt}\left(\mathscr{X}_{1:n}|x_0, o_{1:n}\right)$

While Algorithm 1 is already much more efficient than a straightforward implementation of Equation (13), there is still room for improvement. If $\mathrm{Pos}_k(x_{k-1}) \neq \mathscr{X}_k$, then by Equation (24), we know that $\mathrm{cand}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right)$ has a number of elements that is exponential in the length of the considered sequences, making it very inefficient to execute the steps in Lines 8 and 9 of Algorithm 1. In order to circumvent this problem, we propose a method that does not require an explicit check of the inequality in Criterion (33) for all elements of $\mathrm{cand}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right)$. The approach is identical to that of Algorithm 1, except for Lines 8 and 9, which are replaced by Lines 8' and 9', as given in Algorithm 2.

**Algorithm 2:** An efficient alternative to Lines 8 and 9 of Algorithm 1

...
    ...
        ...
8'         **for** $\hat{x}_k \in \mathscr{X}_k$ **do**
9'             **if** $\alpha_k^{\max}(\hat{x}_k) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$ **then** `Recur`$(\hat{x}_k, k)$
...

In order to be able to define the recursive procedure `Recur` that is used in Line 9' of Algorithm 2, we need some additional notation. First of all, for all $k \in \{1, \ldots, n\}$, $s \in \{k, \ldots, n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$, $x_{k:s} \in \mathscr{X}_{k:s}$ and $o_{k:n} \in \mathscr{O}_{k:n}$, we define

$$\mathrm{cand}_{\hat{x}_{k:s}}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right) := \left\{x_{k:n} \in \mathrm{cand}\left(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}\right) : x_{k:s} = \hat{x}_{k:s}\right\}. \tag{35}$$

Secondly, for all $k \in \{1, \ldots, n\}$, $s \in \{k, \ldots, n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$ and $\hat{x}_{k:s} \in \mathscr{X}_{k:s}$, we define $\alpha_k^{\mathrm{opt}}(\hat{x}_{k:s}|x_{k-1})$ as follows. For $s = k$, we let $\alpha_k^{\mathrm{opt}}(\hat{x}_{k:k}|x_{k-1}) := \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$, as given by Equation (32). For

$s \in \{k+1,\dots,n\}$, $\alpha_k^{\text{opt}}(\hat{x}_{k:s}|x_{k-1})$ is then recursively defined by

$$\alpha_k^{\text{opt}}(\hat{x}_{k:s}|x_{k-1}) = \frac{\alpha_k^{\text{opt}}(\hat{x}_{k:s-1}|x_{k-1})}{\overline{S}_{s-1}(\{o_{s-1}\}|\hat{x}_{s-1})\overline{Q}_s(\{\hat{x}_s\}|\hat{x}_{s-1})}. \tag{36}$$

**Procedure** `Recur`$(\hat{x}_{k:s}, s)$

1 **if** $s = n$ **then**
2     add $\hat{x}_{k:n}$ to $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$
3 **else**
4     **for** $\hat{x}_{s+1} \in \mathscr{X}_{s+1}$ **do**
5         **if** $\text{cand}_{\hat{x}_{k:s}\oplus\hat{x}_{s+1}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ **then**
6             **if** $\alpha_{s+1}^{\max}(\hat{x}_{s+1}) \geq \alpha_k^{\text{opt}}(\hat{x}_{k:s} \oplus \hat{x}_{s+1}|x_{k-1})$ **then** `Recur`$(\hat{x}_{k:s} \oplus \hat{x}_{s+1}, s+1)$

The following result establishes that Lines 8' and 9' of Algorithm 2 are indeed a valid alternative for Lines 8 and 9 of Algorithm 1.

**Theorem 4.** *The set* $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ *that is obtained by executing Algorithm 2 is correct, in the sense that it satisfies Equation* (33).

As we will show in Section 7, Algorithm 2 is surprisingly efficient. One of the reasons for this efficiency is that checking the if-conditions in Lines 5 and 6 of the Procedure `Recur` is really easy, perhaps in contrast to what one might think at first sight. For the condition in Line 6, this is because one can use Equation (36) to derive $\alpha_k^{\text{opt}}(\hat{x}_{k:s} \oplus \hat{x}_{s+1}|x_{k-1})$ from $\alpha_k^{\text{opt}}(\hat{x}_{k:s}|x_{k-1})$, the latter of which is either available from the previous call to the Procedure `Recur` or, if $s = k$, equal to $\alpha_k^{\text{opt}}(\hat{x}_k|x_{k-1})$, which has already been calculated during the initial set-up phase (see Section 6.1). Before we can explain why checking the condition in Line 5 is easy as well, we first need to introduce the data structure that we use to store the sets $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ of optimal sequences.

For $k = n$, $\text{opt}\,(\mathscr{X}_n|x_{n-1}, o_n)$ is simply a list of states $\hat{x}_n \in \mathscr{X}_n$. For $k < n$, we could also just store the optimal sequences $\hat{x}_{k:n}$ in $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ as a simple list, but this would imply storing the same information multiple times, since the initial part of some of those sequences will be identical. Furthermore, it would make checking the condition in Line 5 of the Procedure `Recur` very elaborate. We therefore choose to represent the set $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ as a collection of trees. Each $\hat{x}_k \in \mathscr{X}_k$ that satisfies the inequality in Line 9' corresponds to a root of a tree. The paths of these trees correspond to elements of $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$.

**Example 2.** We consider a simple binary HMM with, for all $i \in \{1,\dots,n\}$, $\mathscr{X}_i = \{0,1\}$. Then for $k = n-7$, we could for example find that

$$\text{opt}\,(\mathscr{X}_{k:n}|\,0, o_{k:n}) = \{00001000, 00001010, 00001110, 00011110, 10001010, 10001110\}.$$

That same set of optimal sequences can also be represented as a collection of trees, which is depicted in Figure 5. ♦

Representing $\text{opt}\,(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ as a collection of trees has two important advantages. The first advantage is that such a collection of trees can be constructed step by step while running Algorithm 2. In Line 9' of that algorithm, with every call to the Procedure `Recur`, we add the current

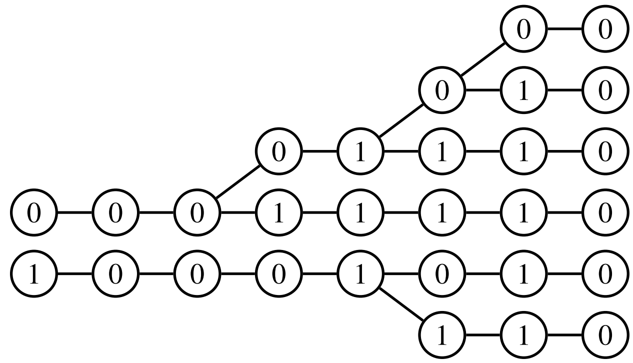

Figure 5: Tree representation of $\operatorname{opt}(\mathscr{X}_{k:n}|\,0, o_{k:n})$, for $k = n-7$

state $\hat{x}_k$ as a root node of a new tree. With every subsequent recursive call to the Procedure Recur (in Line 6 of that same procedure), we add a new child $\hat{x}_{s+1}$ to an already existing node $\hat{x}_s$, where $\hat{x}_s$ is the last state of the presently considered sequence $\hat{x}_{k:s}$.

In order for such a step by step construction to lead to a representation for $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$, each path of the resulting set of trees must have length $n-k+1$. In other words, it is necessary that every node in this representation has at least one child, except for nodes that form the end of a path that has length $n-k+1$. Equivalently, and more technically, it is necessary that with every execution of Line 4 of the Procedure Recur, at least one $\hat{x}_{s+1} \in \mathscr{X}_{s+1}$ satisfies both of the subsequent if-conditions (in Lines 5 and 6). The following result establishes that this condition is always met.

**Theorem 5.** *Fix $k \in \{1, \ldots, n-1\}$ and $s \in \{k, \ldots, n-1\}$ and consider any execution of the Procedure* Recur$(\hat{x}_{k:s}, s)$ *while running Algorithm 2. Then there will be at least one $x_{s+1} \in \mathscr{X}_{s+1}$ for which we obtain that both* $\operatorname{cand}_{\hat{x}_{k:s} \oplus x_{s+1}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ *[the if-condition in Line 5] and $\alpha^{\max}_{s+1}(x_{s+1}) \geq \alpha^{\mathrm{opt}}_k(\hat{x}_{k:s} \oplus x_{s+1}|x_{k-1})$ [the if-condition in Line 6].*

All that is now left to explain is how the if-condition in Line 5 of the Procedure Recur can be checked efficiently. We consider two distinct cases: $\hat{x}_k \in \operatorname{Pos}_k(x_{k-1})$ and $\hat{x}_k \notin \operatorname{Pos}_k(x_{k-1})$. If $\hat{x}_k \notin \operatorname{Pos}_k(x_{k-1})$, then by Equations (24) and (35), we find that

$$\operatorname{cand}_{\hat{x}_{k:s} \oplus \hat{x}_{s+1}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) = \hat{x}_{k:s} \oplus \hat{x}_{s+1} \oplus \mathscr{X}_{s+2:n} \neq \emptyset,$$

which makes the if-condition in Line 5 trivially true. If $\hat{x}_k \in \operatorname{Pos}_k(x_{k-1})$, then again by Equations (24) and (35), $\operatorname{cand}_{\hat{x}_{k:s} \oplus \hat{x}_{s+1}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ if and only if $\operatorname{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n})$ contains a sequence that starts with $\hat{x}_{k+1:s} \oplus \hat{x}_{s+1}$. If we represent $\operatorname{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n})$ as a collection of trees, this is equivalent to checking whether $\hat{x}_{s+1}$ is a child of the node that corresponds to the last state in the sequence $\hat{x}_{k:s}$.

This brings us to the second advantage of representing sets of optimal sequences as a collection of trees: it makes checking the if-condition in Line 5 of the Procedure Recur both elegant and efficient. If in Line 8' of Algorithm 2, $\hat{x}_k \notin \operatorname{Pos}_k(x_{k-1})$, then in all of the subsequent calls to the Procedure Recur, Line 5 can simply be ignored. If $\hat{x}_k \in \operatorname{Pos}_k(x_{k-1})$, then in all the subsequent calls to the Procedure Recur, Lines 5 and 6 can be condensed into a single for-loop that runs over the children of the node that corresponds to $\hat{x}_s$. Hence, for $\hat{x}_k \in \operatorname{Pos}_k(x_{k-1})$, executing Line 8' of Algorithm 2—including all the subsequent recursive calls to the Procedure Recur—can be done efficiently by traversing depth-first through the trees that make up the representation of $\mathrm{opt}\,(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n})$, copying them as we go, except for those nodes for which the condition in Line 6 of the Procedure `Recur` is not satisfied.

**Example 3.** We continue with Example 2. This time, we let $k = n-8$, which implies that Figure 5 is now a representation for $\mathrm{opt}\,(\mathscr{X}_{k+1:n}|\,0, o_{k+1:n})$. For the sake of this example, let us assume that $0 \in \mathrm{Pos}_k(0)$ and $1 \notin \mathrm{Pos}_k(0)$. Then we know from Equations (23) and (24) that every sequence $\hat{x}_{k:n}$ in $\mathrm{opt}\,(\mathscr{X}_{k:n}|\,0, o_{k:n})$ is an element of either $0 \oplus \mathrm{opt}\,(\mathscr{X}_{k+1:n}|\,0, o_{k+1:n})$ or $1 \oplus \mathscr{X}_{k+1:n}$. Hence, $\mathrm{opt}\,(\mathscr{X}_{k:n}|\,0, o_{k:n})$ might for example have a representation that looks like the one depicted in Figure 6.

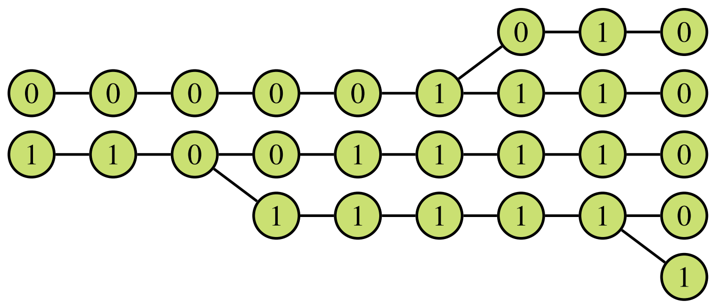

Figure 6: Tree representation of $\mathrm{opt}\,(\mathscr{X}_{k:n}|0, o_{k:n})$, for $k = n-8$

Figure 7 should clarify how Algorithm 2 constructs this representation. The two sequences in $\mathrm{opt}\,(\mathscr{X}_{k:n}|\,0, o_{k:n})$ that start with 0 correspond to the green—grey in monochrome versions of this paper—branches in the topmost part of Figure 7. The first node of these two sequences was added in Line 9' of Algorithm 2 (in this case, the if-condition in that line turned out to be true). The other green nodes in the topmost part of Figure 7 were added during the subsequent calls to the Procedure `Recur`. Checking the if-condition in Line 5 of those procedures was done by traversing depth-first through the nodes of the representation of $\mathrm{opt}\,(\mathscr{X}_{k+1:n}|\,0, o_{k+1:n})$. The red—grey with a thicker outline in monochrome versions of this paper—nodes correspond to states for which the if-condition in Line 6 was not satisfied. The (white) descendants of these red nodes were never visited by the algorithm; we depict them only to allow for an easy comparison with Figure 5. The tree representation for the three sequences in $\mathrm{opt}\,(\mathscr{X}_{k:n}|\,0, o_{k:n})$ that start with 1 were constructed in a similar manner, which is illustrated in the bottommost part of Figure 7. The main difference is that for those sequences, since $1 \notin \mathrm{Pos}_k(0)$, the if-condition in Line 5 of the Procedure `Recur` is trivially true, implying that the algorithm does not need to traverse through a tree representation of $\mathrm{opt}\,(\mathscr{X}_{k+1:n}|\,1, o_{k+1:n})$, but rather trough the complete set $\mathscr{X}_{k+1:n}$. Again, we stop whenever the if-condition on Line 6 is not satisfied, as symbolised by the red nodes. ♦

### 6.3 Additional Comment

It might happen that the available information consists of assessments for the lower and upper transition and emission probabilities only:

$$\underline{Q}_k(\{x_k\}|x_{k-1}),\ \overline{Q}_k(\{x_k\}|x_{k-1}),\ \underline{S}_k(\{o_k\}|x_k) \text{ and } \overline{S}_k(\{o_k\}|x_k)$$

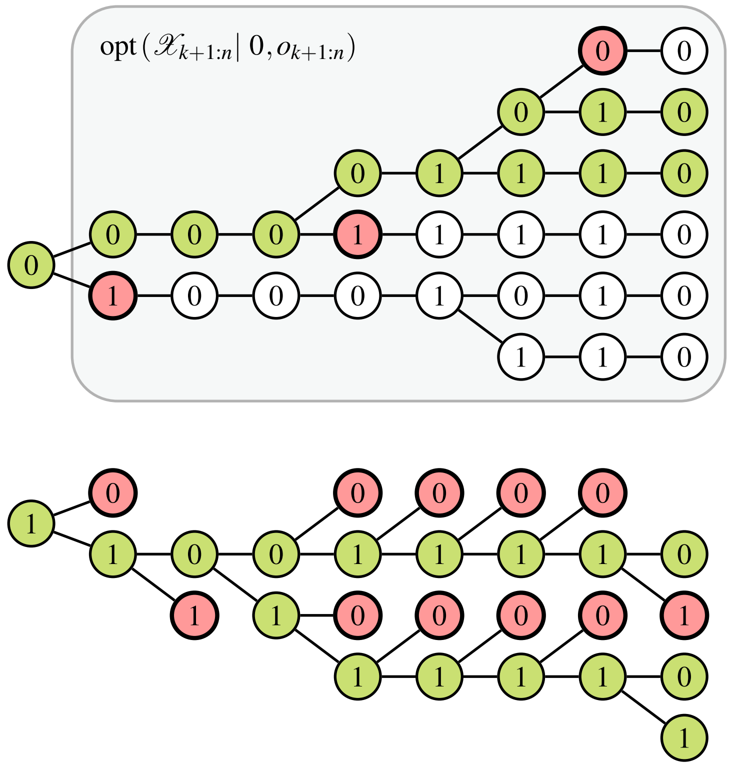


Figure 7: Clarification of the construction of opt $(\mathscr{X}_{k:n}|0, o_{k:n})$, for $k = n - 8$

for all $k \in \{1, \ldots, n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$, $x_k \in \mathscr{X}_k$ and $o_k \in \mathscr{O}_k$. In that case, one can use the following method to construct, for all $k \in \{1, \ldots, n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$ and $x_k, \hat{x}_k \in \mathscr{X}_k$ such that $x_k \neq \hat{x}_k$, a conservative value for the threshold $\theta_k(\hat{x}_k, x_k | x_{k-1})$.

The most conservative coherent models $\underline{Q}_k(\cdot|X_{k-1})$ that correspond to assessments of lower and upper probabilities of singletons are 2-monotone (de Campos, Huete, & Moral, 1994). Due to their comonotone additivity (De Cooman, Troffaes, & Miranda, 2008), this implies that:

$$\underline{Q}_k(\mathbb{I}_{\{x_k\}} - a\mathbb{I}_{\{\hat{x}_k\}} | x_{k-1}) = \underline{Q}_k(\{x_k\} | x_{k-1}) - a\overline{Q}_k(\{\hat{x}_k\} | x_{k-1})$$

for all $a \geq 0$, and therefore Equation (27) leads to

$$\theta_k(\hat{x}_k, x_k | x_{k-1}) = \frac{\underline{Q}_k(\{x_k\} | x_{k-1})}{\overline{Q}_k(\{\hat{x}_k\} | x_{k-1})}.$$

The right-hand side is the smallest possible value of the threshold $\theta_k(\hat{x}_k, x_k | x_{k-1})$ corresponding to the assessments $\underline{Q}_k(\{x_k\} | x_{k-1})$ and $\overline{Q}_k(\{\hat{x}_k\} | x_{k-1})$, leading to the most conservative inferences and therefore the largest possible sets of maximal sequences that correspond to these assessments.

## 7. Discussion of the Algorithm's Complexity

This section discusses the computational compexity of the different steps of the EstiHMM algorithm, as developed in the previous section. In the end, we find that the total complexity of the EstiHMM algorithm is polynomial in the size of the input—quadratic in the length of the iHMM and cubic in the number of states—, as well as linear in the size of the output—the number of maximal sequences in $\mathrm{opt}(\mathscr{X}_{1:n}|o_{1:n})$. The linearity in the size of the output is especially interesting; we discuss this in Section 7.4.

### 7.1 Preparatory Calculations

We begin with the preparatory calculations of the quantities in Equations (27)–(32). For the thresholds $\theta_k(\hat{x}_k, x_k|z_{k-1})$ in Equation (27), the computational complexity is clearly cubic in the number of states, and—except for stationary iHMMs—linear in the length of the iHMM. Calculating the $\alpha_k^{\max}(x_k)$ and $\beta_k^{\max}(x_k)$ in Equations (29) and (30) is linear in the length of the iHMM—even for stationary iHMMs—and quadratic in the number of states. The complexity of finding the $\alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$ in Equation (32) is therefore—in the worst, non-stationary case—linear in the lenght of the iHMM and cubic in the number of states.

### 7.2 Algorithm 2

The computational complexity of Algorithm 2 is less trivial. Let us start by noting that this construction essentially consists in repeating the same small step over and over again, namely executing the Procedure `Recur`. As we explained in the previous section, our data structure—a collection of trees—enables us to do this very efficiently. Each of the three if-conditions in the Procedure `Recur` can be checked in constant time. Therefore, taking into account the for-loop in Line 4, we find that the computational complexity of a single execution of the Procedure `Recur` is linear in the number of states.

Next, notice that every optimal sequence $\hat{x}_{k:n}$ that is obtained by running Algorithm 2 is constructed by adding extra states $\hat{x}_{s+1}$ to an already constructed sequence $\hat{x}_{k:s}$, repeating this for $s$ going from $k$ to $n-1$. Adding such a state means executing the Procedure `Recur` once, and is therefore linear in the number of states. Similarly, creating the first state $\hat{x}_k$ is at most linear in the number of states as well—due to the for-loop in Line 8' of Algorithm 2. Hence, constructing a single optimal sequence $\hat{x}_{k:n}$ is linear in the length of this sequence, as well as linear in the number of states. By Theorem 5, we also know that every execution of the Procedure `Recur` is guaranteed to be part of the construction of at least one optimal sequence. Therefore, we find that constructing a single set $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$—executing Algorithm 2—is linear in the number of optimal sequences it consists of, linear in the length of those sequences and linear in the number of states.

### 7.3 Algorithm 1

What Algorithm 1 basically does to obtain the set $\mathrm{opt}(\mathscr{X}_{1:n}|o_{1:n})$ is to construct all of the the sets $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$, for every $x_{k-1} \in \mathscr{X}_{k-1}$, letting $k$ run from $n$ to 1. For $k = n$ and a fixed $x_{n-1} \in \mathscr{X}_{n-1}$, this is linear in the number of states—see Lines 3 and 4 of Algorithm 1. For $k < n$ and a fixed $x_{k-1} \in \mathscr{X}_{k-1}$, this comes down to executing Algorithm 2. As shown in the previous section, Algorithm 2 is linear in the number of optimal sequences in $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$, linear in the length of those sequences $(n-k+1)$ and linear in the number of states. Hence, we conclude that complexity of Algorithm 1 is quadratic in the length of the iHMM, quadratic in the number of states and roughly speaking[6] linear in the number of maximal sequences.

### 7.4 The Total Complexity

The complete EstiHMM algorithm consists of the preparatory calculations in Section 6.1 and a single execution of Algorithm 1, where, in the latter, Lines 8 and 9 are replaced by their more efficient versions in Algorithm 2. We conclude from the previous sections that the total computational complexity of all of this is—at worst—quadratic in the length of the iHMM, cubic in the number of states, and roughly speaking linear in the number of maximal sequences.

This linearity in the number of maximal sequences is clearly the remaining bottleneck of the algorithm, since there may be exponentially many such sequences. However, this should not lead the reader to conclude that the EstiHMM algorithm has exponential complexity, meaning that it is exponential in the size of the *input*—the length of the iHMM and the number of states. It is crucial to realise that the complexity is linear in the size of the *output*—the number of maximal sequences—, which in turn may be exponential in the input. However, as long as the size of the output is bounded, the algorithm is guaranteed to have a computational complexity that is polynomial in the size of the input. No such guarantee can be given for algorithms whose complexity is linear in the size of the input—for example a naive implementation of Algorithm 1 that does not replace Lines 8 and 9 by their more efficient counterparts in Algorithm 2.

Although linearity in the size of the output might seem rather bad, it is in fact all we can hope for. Even simply printing the output—all maximal sequences—already has a computational complexity that is linear in its size as well as linear in the length of the iHMM. Linearity in the size of the output is inherent to all problems that do not necessarily lead to a single solution, but allow for set-valued solutions as well. If the size of the output is too large, then no algorithm, however cleverly designed, can overcome this hurdle.

In order for the EstiHMM algorithm not to choke when the number of maximal sequences is very large, one can keep track—for every set $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$—of how many times Line 2 of the Procedure `Recur` has been executed so far, aborting the algorithm whenever some preset treshold has been exceeded. It is however not possible to return only the $k$ best solutions, simply because there is no such thing as a better or worse maximal sequence; they are all incomparable. The only way in which the number of maximal sequences can be reduced is by decreasing the imprecision of the model: to gather extra data or expert knowledge, leading to smaller local credal sets, pointwise larger local lower previsions and therefore fewer maximal sequences. Alternatively, one could also consider using E-admissable sequences—of which there may be multiple as well, but not as many as maximal ones—or Γ-maximin sequences—of which, in most instances, there is only one. However, we know of no algorithm that can calculate the E-admissable or Γ-*maximin* sequences in an efficient manner, let alone one that is linear in the output; see Section 4.3.

### 7.5 Comparison with Viterbi's Algorithm

For precise HMMs, the state sequence estimation problem can be solved very efficiently by the Viterbi algorithm (Rabiner, 1989; Viterbi, 1967), whose complexity is linear in the length of the HMM, and quadratic in the number of states. However, this algorithm only emits a single optimal—

6. For every $k$ and $x_{k-1} \in \mathscr{X}_{k-1}$, constructing the set $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1},o_{k:n})$ has linear complexity in the number of optimal sequences at that stage.

most probable—state sequence, even in cases where there are multiple—equally probable—optimal solutions: this of course simplifies the problem. If we would content ourselves with giving only a single maximal solution, the ensuing version of our algorithm would have a complexity that is similar to Viterbi's.

So, to allow for a fair comparison between Viterbi's algorithm and ours, we would need to alter Viterbi's algorithm in such a way that it no longer resolves ties arbitrarily, and emits all—equally probable—optimal state sequences. This new version will remain linear in the length of the HMM, and quadratic in the number of states, but will also have added complexity. As discussed in the previous section, even printing all optimal sequences is linear in the number of them and therefore possibly exponential, for example if all possible solutions are equally probable—imagine a precise HMM of which all local probability mass functions are uniform.

For the complexity of the most time-consuming part of our algorithm—Algorithm 1—, the only difference is this: Viterbi's approach is linear and ours is quadratic in the length of the HMM. Where does this difference come from? In imprecise HMMs we have mutually incomparable solutions, whereas in precise HMMs the optimal solutions are indifferent, or equally probable. This makes sure that the algorithm for precise HMMs requires no forward loops, as is the case in the EstiHMM algorithm, every time we run Algorithm 2. We believe that this added complexity is a reasonable price to pay for the robustness that working with imprecise-probabilistic models offers.

## 8. Some Experiments

Since the complexity of the EstiHMM algorithm depends so crucially on the number of maximal sequences it emits, the present section will study this number in more detail. We do so by taking a closer look at how it depends on the transition probabilities of the model, and how it evolves when we let the imprecision of the local models grow. We shall see that the number of maximal sequences displays very interesting behaviour that can be explained, and even predicted to some extent. To allow for easy visualisation, we limit this discussion to stationary binary iHMMs, where both the state and output variables can assume only two possible values, say 0 and 1.

### 8.1 Describing a Stationary Binary iHMM

The precise transition probabilities for going from one state to the next are completely determined by numbers in the unit interval: the probability $p$ to go from state 0 to state 0, and the probability $q$ to go from state 1 to state 0. To further pin down the HMM we also need to specify the marginal probability $m$ for the first state to be 0, and the two emission probabilities: the probability $r$ of emitting output 0 from state 0 and the probability $s$ of emitting output 0 from state 1.

In this binary case, all coherent imprecise-probabilistic models can be found by contamination: taking convex mixtures of precise models, with mixture coefficient $1-\varepsilon$, and the vacuous model, with mixture coefficient $\varepsilon$, leading to a so-called linear-vacuous model (Walley, 1991), often referred to as an $\varepsilon$-contaminated model as well. To simplify the analysis, we let the emission model remain precise, and use the same mixture coefficient $\varepsilon$ for the marginal and the transition models. As $\varepsilon$ ranges from zero to one, we then evolve from a precise HMM towards an iHMM with vacuous marginal and transition models (and precise emission models).

### 8.2 An iHMM of Length Two

We now examine the behaviour of an iHMM of length two, with the following precise probabilities fixed:

$$m = 0.1,\ r = 0.8 \text{ and } s = 0.3.$$

Fixing an output sequence and a value for $\varepsilon$, we can use our algorithm to calculate the corresponding numbers of maximal state sequences as $p$ and $q$ range over the unit interval. The results can be represented conveniently in the form of a heat plot. The plots in Figure 8 correspond to the output sequence $o_{1:2} = 01$.

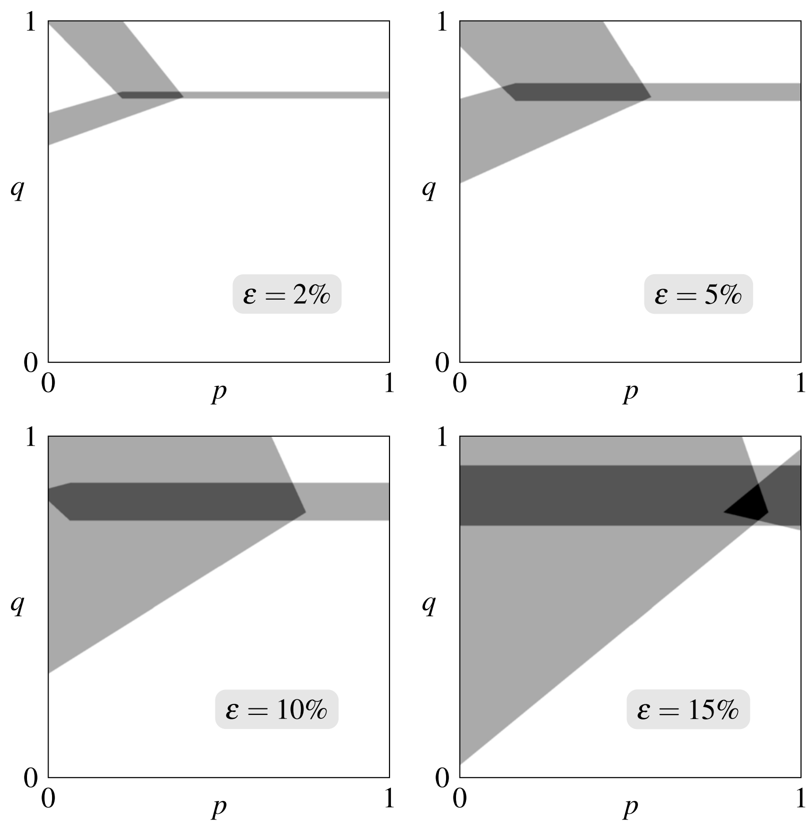


Figure 8: Heat plots for $o_{1:2} = 01$

The number of maximal state sequences clearly depends on the transition probabilities $p$ and $q$. In the rather large parts of 'probability space' that are coloured white, we get a single maximal sequence—as we would for HMMs—, but there are continuous regions where we see a higher number appear. In the present example—a binary chain of length two—, the highest possible number of maximal sequences is of course four. In the dark grey area, there are three maximal sequences, and two in the light grey regions. The plots show what happens when we let $\varepsilon$ increase: the grey areas expand and the number of maximal sequences increases. For $\varepsilon = 15\%$, we even find a small area—coloured black—where all four possible state sequences are maximal: locally, due to the relatively high imprecision of our local models, we cannot provide any useful robust estimate for the state sequence producing the output sequence $o_{1:2} = 01$.

For small $\varepsilon$, the areas with more than one maximal state sequence are quite small and seem to resemble strips that narrow down to lines as $\varepsilon$ tends to zero. This suggests that we should be able to explain at least qualitatively where these areas come from by looking at compatible precise models: the regions where an iHMM produces different maximal (mutually incomparable) sequences, are widened versions of loci of indifference for precise HMMs.

By a *locus of indifference*, we mean the set of $(p,q)$ that correspond to two given state sequences $x_{1:2}$ and $\hat{x}_{1:2}$ having equal posterior probability:

$$p(x_{1:2}|o_{1:2}) = p(\hat{x}_{1:2}|o_{1:2}),$$

or, provided that $p(o_{1:2}) > 0$,

$$p(x_{1:2},o_{1:2}) = p(\hat{x}_{1:2},o_{1:2}).$$

In our example, where $o_{1:2} = 01$, we find the following expressions for each of the four possible state sequences:

$$p(00,01) = mr(1-r)p; \qquad p(10,01) = (1-m)s(1-r)q;$$
$$p(01,01) = mr(1-s)(1-p); \qquad p(11,01) = (1-m)s(1-s)(1-q).$$

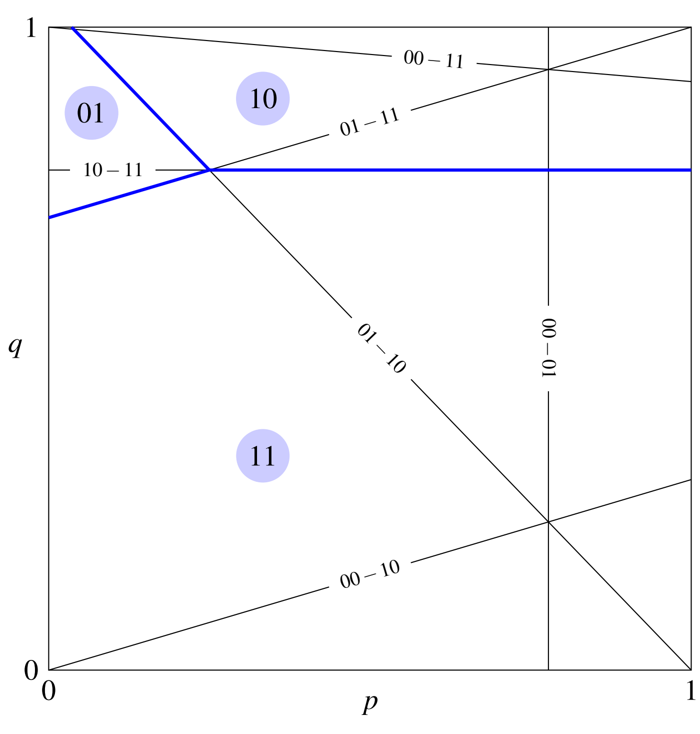


Figure 9: Loci of indifference for $o_{1:2} = 01$

By equating any two of these expressions, we express that the corresponding two state sequences have an equal posterior probability. Since the resulting equations are a function of $p$ and $q$ only, each of these six possible combinations defines a locus of indifference. All of them are depicted as lines in Figure 9.

111000001101000001000100001111111110111101000010110110110000 ...

111000001101000001000100001111111110111101000010110110110000 ...
11**0**000001101000001000100001111111110111101000010110110110000 ...
11100000**0**101000001000100001111111110111101000010110110110000 ...
11100000110**0**000001000100001111111110111101000010110110110000 ...
11100000110100000**0**000100001111111110111101000010110110110000 ...

Figure 10: Maximal sequences for an iHMM of length 100

Parts of these loci, depicted in blue—darker and bolder in monochrome versions of this paper—, demarcate the three regions where the state sequences 01, 10 and 11 are optimal—have the highest posterior probability.

What happens when the transition models become imprecise? Roughly speaking, nearby values of the original $p$ and $q$ enter the picture, effectively turning the loci—lines—of indifference into bands of incomparability: the emergence of regions with two and more maximal sequences can be seen to originate from the loci of indifference; compare Figure 9 with Figure 8.

### 8.3 An iHMM of Length 100

In order to demonstrate that our algorithm is indeed efficient, we let it determine the maximal sequences for a random output sequence of length 100.

We consider the same stationary binary HMM as before, but with the following precise marginal and emission probabilities:

$$m = 0.1,\ r = 0.98, \text{ and } s = 0.01.$$

In practical applications, the probability for an output variable to have the same value as the corresponding hidden state variable is usually quite high, which explains why we have chosen $r$ and $s$ to be close to 1 and to 0, respectively. In contrast with the previous experiments, we do not let the transition probabilities vary, but fix them to the following values:

$$p = 0.6 \text{ and } q = 0.5.$$

The local models of the iHMM that we use to determine the maximal sequences are now generated by $\varepsilon$-contaminating these precise local models. We use the same mixture coefficient $\varepsilon$ for the marginal, transition and emission models. In Figure 10, we show the five maximal sequences that correspond to the highlighted output sequence, with $\varepsilon = 2\%$. Due to space constraints, we display only the first 60 digits of these sequences. Since the emission probabilities were chosen to be quite accurate, it is no surprise that the output sequence itself is one of the maximal sequences. In addition, we have indicated in bold face the state values that differ from the outputs in the output sequence; in the 40 digits that are not displayed, no such differences occured. We see that the model represents more indecision about the values of the state variables as we move further away from the end of the sequence. This is a result of a phenomenon called *dilation*, which—as has been noted in another paper (De Cooman et al., 2010)—tends to occur when inferences in a credal tree proceed from the leaves towards the root.

As for the efficiency of our algorithm: it took about 0.2 seconds to calculate these 5 maximal sequences.[7] The reason why this could be done so fast is that the algorithm is more or less linear

7. Running a Python program on a 2012 MacBook Pro.

in the number of solutions (see Section 7), which in this case is only 5. If we let $\varepsilon$ grow to for example 5%, the number of maximal sequences for the same output sequence is 764 and these can be determined in about 32 seconds. This demonstrates that the complexity is indeed more or less proportional to—and therefore linear in—the number of solutions and that the algorithm can efficiently calculate the set of maximal sequences, even for long output sequences. For larger values of $\varepsilon$, say 10%, it took more than 30 minutes to determine all maximal sequences, leading us to abort the algorithm. This should not lead the reader to conclude that for large $\varepsilon$, the EstiHMM algorithm is no longer linear in the number of maximal sequences. No, it simply means that—at least for long iHMMs—this number of maximal sequences can increase quickly as soon as $\varepsilon$ passes some critical boundary.

## 9. An Application in Optical Character Recognition

As a first application, we use the EstiHMM algorithm to detect and correct mistakes in words. The hidden sequence $x_{1:n}$ corresponds to the original, correct version of a word, of which the output sequence $o_{1:n}$ is an artificially corrupted version. In this way, we simulate observational processes that are not perfectly reliable, such as the output of an Optical Character Recognition (OCR) device. This leads to observed output sequences that may contain errors, which we will try to detect and correct. The original words were taken from Dante's *Divina Commedia*, of which the 1018 words of the second canto were used as a training set and the initial 200 words of the first canto as a test set. By comparing the results of the EstiHMM algorithm with those of the Viterbi algorithm, we are able to illustrate some of the advantages of the former.

### 9.1 Learning the Local Models

In order to apply our algorithm, we must identify a local uncertainty model for each original and observed letter: a marginal model $\underline{Q}_1$ for the first letter $X_1$ of the original word, a transition model $\underline{Q}_k(\cdot|X_{k-1})$ for the subsequent letters $X_k$, with $k \in \{2,\ldots,n\}$, and an emission model $\underline{S}_k(\cdot|X_k)$ for the observed letters $O_k$, with $k \in \{1,\ldots,n\}$. We use the same state space $\mathscr{X} = \mathscr{O}$ for all these variables, consisting of the 21 letters of the Italian alphabet. For the sake of simplicity, we assume stationarity, making the transition and emission models independent of $k$.

For the identification of the local models of the iHMM, we use the imprecise Dirichlet model (IDM) (Walley, 1996). This corresponds to considering the set of all Dirichlet priors with some fixed strength $s > 0$, using the lower and upper bounds of the inferences obtained by each of these priors as our model. For example, for the marginal model $\underline{Q}_1$, applying the IDM leads to the following lower and upper probabilities:

$$\underline{Q}_1(\{x\}) = \frac{n_x}{s + \sum_{z \in \mathscr{X}} n_z} \text{ and } \overline{Q}_1(\{x\}) = \frac{s + n_x}{s + \sum_{z \in \mathscr{X}} n_z} \text{ for all } x \in \mathscr{X},$$

where, for all $z \in \mathscr{X}$, $n_z$ is the number of words in the training text for which the first letter $X_1$ is equal to $z$. The hyperparameter $s$ can be regarded as a degree of caution that is taken into account in the inferences. We use $s = 2$; Walley (1996, Section 2.5) provides a number of arguments in favour of this choice. For the transition and emission models, we can proceed similarly, by counting the transitions of one letter to another, respectively in the original word or during the observation process. In this way, we obtain lower and upper transition and emission probabilities for singletons, which, as pointed out in Section 6.3, suffice to run the algorithm. In fact, since the IDM leads to local models that are linear-vacuous and hence completely and therefore also 2-monotone, the approach described in Section 6.3 actually leads to exact values for the parameters $\theta_k(\hat{x}_k, x_k|x_{k-1})$ instead of a conservative approximation.

For the identification of the local models of the precise HMM, we use a similar but now precise Dirichlet model approach, with a Perks's prior that has the same prior strength $s = 2$. As an example, for the precise marginal model $Q_1$, this leads to the following simple identification:

$$Q_1(\{x\}) = \frac{s/|\mathscr{X}| + n_x}{s + \sum_{z \in \mathscr{X}} n_z},$$

where $|\mathscr{X}|$ is the number of states.

The difference between the precise and imprecise models that are constructed in the way described above is relatively small. For example, using our training set of 1018 words, 67 of which start with the letter A, we obtained the following (lower, upper and precise) probability that the first letter of a word is A:

$$\underline{Q}_1(\{\mathrm{A}\}) = 0.06569,\ Q_1(\{\mathrm{A}\}) = 0.06578 \text{ and } \overline{Q}_1(\{\mathrm{A}\}) = 0.06765.$$

Nevertheless, as illustrated in the next section, the imprecise model can lead to results that are rather diferent from those obtained by the precise model.

### 9.2 Results

Let us first discuss an example of the difference between the results obtained by the Viterbi and the EstiHMM algorithm, in order to illustrate an important advantage of the latter. OCR software has mistakenly read the Italian word QUANTO as OUANTO. Using a precise model, the Viterbi algorithm does not correct this mistake, as it suggests that the original correct word is DUANTO. The EstiHMM algorithm on the other hand, using an imprecise model, returns CUANTO, DUANTO, FUANTO and QUANTO as maximal, undominated solutions, including the correct one. Of course we would still have to pick the correct solution out of this set of suggestions—for example by using a dictionary or a human opinion—, but by using the EstiHMM algorithm, we have managed to reduce the search space from all possible five letter words to the much smaller set of four words given above. Notice that the solution of the Viterbi algorithm is included in the maximal solutions EstiHMM returns. One can easily prove that this will always be the case.

We applied our method to the first 200 words of the first *canto* of Dante's *Divina Commedia*, 137 of which where correctly read by our artifical OCR device and 63 of which contained errors. We tried to correct these errors using both the EstiHMM and the Viterbi algorithm, and compare both approaches. The results are summarised in Table 1.

For the Viterbi algorithm, the main conclusion is that applying it to the output of the OCR device results in a decreased number of incorrect words. The number of correct words rises from 68.5% to 78.5%. However, the Viterbi algorithm also introduces new errors for 5 correctly read words.

The EstiHMM algorithm manages to suggest the original correct word as one of her solutions in 86% of the cases. Assuming we are able to detect this correct word, the percentage of correct words rises from 68.5% to 86% by applying the EstiHMM algorithm, thereby outperforming the Viterbi algorithm by almost 10%. Secondly, we also notice that the EstiHMM algorithm has never introduced new errors in words that were already correct.

| | *total number* | *correct after OCR* | *wrong after OCR* |
|---|---|---|---|
| *total number* | 200 (100%) | 137 (68.5%) | 63 (31.5%) |
| **Viterbi** | | | |
| *correct solution* | 157 (78.5%) | 132 | 25 |
| *wrong solution* | 43 (21.5%) | 5 | 38 |
| **EstiHMM** | | | |
| *correct solution included* | 172 (86%) | 137 | 35 |
| *correct solution not included* | 28 (14%) | 0 | 28 |

Table 1: Summary of the results of the EstiHMM and Viterbi algorithm

Of course, since the EstiHMM algorithm allows for multiple solutions, instead of a single one, it is no surprise that we manage to increase the amount of times we suggest the correct solution. This would happen even if we added random extra solutions to the solution of the Viterbi algorithm. Giving extra solutions can only be seen as an improvement if this is done smartly. To investigate this, we distinguish between the cases where the EstiHMM algorithm returns a single solution, and those where it returns multiple solutions; and look at how the Viterbi and EstiHMM algorithms compare in those two cases.

The EstiHMM algorithm returned a single solution for 155 of the 200 words. As we have already mentioned above, this single solution will always coincide with the one given by the Viterbi algorithm. The results for the EstiHMM and Viterbi algorithms are summarised in Table 2.

| **EstiHMM (single solutions)** | *total number* | *correct after OCR* | *wrong after OCR* |
|---|---|---|---|
| *total number* | 155 (100%) | 129 (83.2%) | 26 (16.8%) |
| *single correct solution* | 134 (86.5%) | 129 | 5 |
| *single wrong solution* | 21 (13.5%) | 0 | 21 |

Table 2: The instances where EstiHMM produces a single estimate

The percentage of words correctly read by the OCR software is now 83.2% instead of the global 68.5%. When the result of the EstiHMM algorithm is a single solution, this serves as an indication that the word we are trying to correct has a fairly high probability of already being correct. We also see that the eventual percentage of correct words is 86.5%, which is only a slight improvement over the 83.2% that were already correct before applying the algorithms.

Next, we look at the remaining 45 words, for which the EstiHMM algorithm returns more than one maximal element. In this case, we do see a significant difference between the results of the Viterbi and the EstiHMM algorithm because the Viterbi algorithm always returns only a single solution. The results for both algorithms are listed in Table 3.

A first and very important conclusion to be drawn from this table is that if the EstiHMM algorithm is indecisive, this serves as a rather strong indication that the word we are applying the algorithm to does indeed contain errors: when the EstiHMM algorithm returns multiple solutions, the original word has been incorrectly read by the OCR software in 82.2% of cases.

| | *total number* | *correct after OCR* | *wrong after OCR* |
|---|---|---|---|
| *total number* | 45 (100%) | 8 (17.8%) | 37 (82.2%) |
| **EstiHMM (multiple solutions)** | | | |
| *correct solution included* | 38 (84.4%) | 8 | 30 |
| *correct solution not included* | 7 (15.6%) | 0 | 7 |
| **Viterbi** | | | |
| *correct solution* | 23 (51.1%) | 3 | 20 |
| *wrong solution* | 22 (48.9%) | 5 | 17 |

Table 3: The instances where EstiHMM produces a set-valued estimate

A second conclusion, related to the first, is that if the EstiHMM algorithm is indecisive, this also serves as an indication that the result returned by the Viterbi algorithm is less reliable: the percentage of correct words after applying the Viterbi algorithm has dropped to 51.1%, in contrast with the global percentage of 78.5%. The EstiHMM algorithm, however, still gives the correct word as one of its solutions in 84.4% of cases, which is almost as high as its global percentage of 86%. If the set given by the EstiHMM algorithm contains the correct solution, the Viterbi algorithm manages to pick this correct solution out of the set in 60.5% of cases. We see that the EstiHMM algorithm seems to notice that we are dealing with more difficult words and therefore gives us multiple solutions, between which it cannot decide.

### 9.3 Advantages of the Imprecise Approach

We learn from our experiments that the EstiHMM algorithm can be usefully applied to make the results of the Viterbi algorithm more robust, and to gain an appreciation of where it is likely to go wrong. If the EstiHMM algorithm is indeterminate, this serves as an indication of robustness issues that would occur if we solved the same problem with the Viterbi algorithm. In those instances, the EstiHMM algorithm returns multiple solutions, between which it cannot decide, whereas the Viterbi algorithm will pick one out of this set in a fairly arbitrary way—depending on the choice of the prior—, thereby increasing the amount of errors made.

This leads us to conclude that the imprecise approach of the EstiHMM algorithm has two main advantages. The first advantage is that it can easily detect when the precise approach becomes sensitive to the adopted prior: this kind of sensitivity occurs exactly in those instances where the EstiHMM algorithm returns an indeterminate result. The second advantage is that, instead of simply detecting this sensitivity to the choice of prior, the EstiHMM algorithm also offers an alternative solution that does not suffer from such issues, in the form of a set of maximal sequences—a set of suggestions for the correct hidden word. As illustrated by our experiments, this set will often contain the actual correct word.

Future work could try to exploit these set-valued solutions by trying to pick the correct word out of the given set of options in some non-arbitrary way. This could for example be done by comparing the options with the entries of a dictionary. Alternatively, one could consider asking the user for feedback, asking him to choose among the options. In this way, additional data is gathered that can be used to build a better model that is less sensitive to the choice of the prior.

## 10. Conclusions

Interpreting the graphical structure of an imprecise hidden Markov model as a credal network under epistemic irrelevance leads to an efficient algorithm for finding the maximal, undominated hidden state sequences for a given observed sequence. An interesting feature of this algorithm is that it has a computational complexity that is linear in the size of the output—the number of maximal state sequences. Preliminary simulations show that, even for long iHMMs of which the transition models have non-negligible imprecision, this number of maximal state sequences is often reasonably low. It remains to be seen whether this observation can be corroborated by a deeper theoretical analysis.

Our application in OCR clearly shows that the EstiHMM algorithm is able to robustify the results of the Viterbi algorithm. Not only does it reduce the amount of wrong conclusions by providing extra possible solutions, it does so in an intelligent manner. It adds extra solutions in the specific cases where the Viterbi algorithm is likely to be wrong, thereby also serving as an indicator of the reliability of the result given by the Viterbi algorithm. Since these set-valued solutions often contain the correct hidden state sequence, they can be usefully applied in a postprocessing phase, for example by offering the set to the user, asking him for feedback.

A first important avenue of future research would be to compare the EstiHMM algorithm with other methods that also try to robustify the Viterbi algorithm by producing set-valued solutions. We distinguish between two different approaches.

On the one hand, we have imprecise methods such as the one adopted by us. They combine an imprecise model with an imprecise-probabilistic decision criterion. In this paper, we have chosen to use maximality as a decision criterion. However, other decision criteria can be adopted as well; see Section 4.3. Some of these other criteria, such as E-admissibility, also lead to set-valued estimates. A common feature of all of these methods is that they take into account *model uncertainty*: what happens with inferences when the model is imperfect? What happens if instead of a single probability mass function, there are a set of possible candidates? In many instances, the resulting inferences will still be determinate. Set-valued solutions are typically obtained only for those instances where the precise-probabilistic approach is more likely to be wrong.

On the other hand, precise models may lead to set-valued solutions as well. In the context of HMMs, the most important example seems to be the $k$-best Viterbi algorithm (Brown & Golod, 2010). Instead of returning only the a posteriori most probable hidden state sequence, the $k$-best Viterbi algorithm returns the $k$ most probable hidden state sequences. There are two important differences with the imprecise approaches described above. First of all, the $k$-best approach has nothing to do with model uncertainty. Instead, it deals with the *probabilistic uncertainty* that is inherent to the model itself, while assuming that this model is perfectly known. If the model is indeed correct, then by returning the $k$ most probable sequences, the probability of the correct estimate to be included in this set-valued solution increases, at the expense of losing determinacy. Secondly, and related to the previous difference, the $k$-best method will always return $k$ sequences, regardless of the accuracy of the 1-best approach. In contrast, imprecise approaches are typically able to distinguish between easy and hard cases, producing determinate answers for the former and set-valued answers for the latter. Nevertheless, despite these differences, one gets the impression that the $k$-best method can be used to tackle similar applications as the EstiHMM algorithm. It would be interesting to check whether this is indeed the case, and to compare their respective results. We leave this as a topic for future research.

Another, more theoretical avenue of future research is to investigate the extent to which the ideas presented in this paper can be applied to credal networks other than iHMMs under *epistemic irrelevance*. There are two specific instances where we have concrete ideas on how to proceed. First of all, we have strong reasons to believe that it is possible to derive a similarly efficient algorithm for iHMMs whose graphical structure is interpreted as a credal network under *strong independence* rather then epistemic irrelevance. This could be interesting and relevant, as this more stringent independence condition leads to joint models that are less imprecise, and therefore produce fewer maximal state sequences—although they will be included in our solutions. Secondly, the EstiHMM algorithm demonstrates that efficient inference in credal trees under epistemic irrelevance is not necessarily limited to queries with a *single* target node only. In fact, we believe that it is possible to develop polyonomial time algorithms, capable of solving wide classes of inference problems in credal trees under epistemic irrelevance, thereby extending the results of De Cooman et al. (2010).

## Acknowledgments

Jasper De Bock is a Ph.D. Fellow of the Research Foundation - Flanders (FWO) at Ghent University, and has developed the algorithm described here in the context of his Master's thesis, in close cooperation with Gert de Cooman, who acted as his thesis supervisor. The present article describes the main results of this Master's thesis. Research by De Cooman has been supported by SBO project 060043 of the IWT-Vlaanderen.

The authors would like to thank the anonimous referees of this paper and a previous conference version for their useful, constructive comments. They led to a significant improvement of the current version, most notably regarding its presentation. This paper has also benefitted from discussions with Marco Zaffalon, Alessandro Antonucci, Alessio Benavoli, Cassio de Campos, Erik Quaeghebeur and Filip Hermans. We are grateful to Marco Zaffalon for providing travel funds, which allowed us to visit IDSIA and discuss practical applications.

## Appendix A. Proofs of Main Results

In this appendix, we justify the formulas (6), (7), (15), (16), (17), (33) and (34) and we give proofs for Proposition 1 and Theorems 2–5. We will frequently use terms such as positive, negative, decreasing and increasing. We therefore start by clarifying what we mean by them. For $x \in \mathbb{R}$, we say that $x$ is *positive* if $x > 0$, *negative* if $x < 0$, *non-negative* if $x \geq 0$ and *non-positive* if $x \leq 0$. We call a real-valued function $f$ defined on $\mathbb{R}$:

(i) *increasing* if $(\forall x, y \in \mathbb{R})(x > y \Rightarrow f(x) > f(y))$;

(ii) *decreasing* if $(\forall x, y \in \mathbb{R})(x > y \Rightarrow f(x) < f(y))$;

(iii) *non-decreasing* if $(\forall x, y \in \mathbb{R})(x > y \Rightarrow f(x) \geq f(y))$;

(iv) *non-increasing* if $(\forall x, y \in \mathbb{R})(x > y \Rightarrow f(x) \leq f(y))$.

*Proof of Equation* (6). For all $k \in \{1,\dots,n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$, $x_{k:n} \in \mathscr{X}_{k:n}$ and $o_{k:n} \in \mathscr{O}_{k:n}$ we infer from Equation (5) that

$$\begin{aligned}\underline{P}_k(\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) &= \underline{Q}_k(\underline{E}_k(\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|X_k)|x_{k-1}) \\ &= \underline{Q}_k\left(\sum_{z_k\in\mathscr{X}_k}\mathbb{I}_{\{z_k\}}\underline{E}_k(\mathbb{I}_{\{x_k\}}(z_k)\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|z_k)\middle|x_{k-1}\right) \\ &= \underline{Q}_k(\mathbb{I}_{\{x_k\}}\underline{E}_k(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k)|x_{k-1}).\end{aligned}$$

Since $\underline{E}_k(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k) \geq 0$ by C1, we see that C2 transforms the above into

$$= \underline{Q}_k(\mathbb{I}_{\{x_k\}}|x_{k-1})\underline{E}_k(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k),$$

which can be reformulated as

$$\begin{aligned}&= \underline{Q}_k(\mathbb{I}_{\{x_k\}}|x_{k-1})\underline{S}_k(\mathbb{I}_{\{o_k\}}|x_k)\underline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \\ &= \underline{Q}_k(\{x_k\}|x_{k-1})\underline{S}_k(\{o_k\}|x_k)\underline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k),\end{aligned}$$

if we take into account Equation (4), since $\underline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \geq 0$ by C1.

Repeating these steps again and again eventually yields Equation (6):

$$\underline{P}_k(\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) = \prod_{i=k}^{n}\underline{Q}_i(\{x_i\}|x_{i-1})\underline{S}_i(\{o_i\}|x_i).$$

In the last step, for $k = n$, we have used the equality $\underline{E}_n(\{o_n\}|x_n) = \underline{S}_n(\{o_n\}|x_n)$, which follows from Equation (3). $\square$

*Proof of Equation* (7). For all $k \in \{1,\dots,n\}$, $x_{k-1} \in \mathscr{X}_{k-1}$, $x_{k:n} \in \mathscr{X}_{k:n}$ and $o_{k:n} \in \mathscr{O}_{k:n}$ we infer from conjugacy and Equation (5) that

$$\begin{aligned}\overline{P}_k(\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) &= -\underline{P}_k(-\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) \\ &= -\underline{Q}_k(\underline{E}_k(-\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|X_k)|x_{k-1}) \\ &= -\underline{Q}_k\left(\sum_{z_k\in\mathscr{X}_k}\mathbb{I}_{\{z_k\}}\underline{E}_k(-\mathbb{I}_{\{x_k\}}(z_k)\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|z_k)\middle|x_{k-1}\right) \\ &= -\underline{Q}_k(\mathbb{I}_{\{x_k\}}\underline{E}_k(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k)|x_{k-1}) \\ &= -\underline{Q}_k(-\mathbb{I}_{\{x_k\}}(-\underline{E}_k(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k)))|x_{k-1}).\end{aligned}$$

Since $-\underline{E}_k(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k) = \overline{E}_k(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k) \geq 0$ by conjugacy and Lemma 6, we see that C2 and Equation (2) transform the above into

$$\begin{aligned}&= -\big(-\underline{E}_k(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k)\big)\underline{Q}_k(-\mathbb{I}_{\{x_k\}}|x_{k-1}) \\ &= -\overline{Q}_k(\mathbb{I}_{\{x_k\}}|x_{k-1})\underline{E}_k(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_k),\end{aligned}$$

which can be reformulated as

$$\begin{aligned}&= -\overline{Q}_k(\mathbb{I}_{\{x_k\}}|x_{k-1})\overline{S}_k(\mathbb{I}_{\{o_k\}}|x_k)\underline{P}_{k+1}(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \\ &= \overline{Q}_k(\mathbb{I}_{\{x_k\}}|x_{k-1})\overline{S}_k(\mathbb{I}_{\{o_k\}}|x_k)\overline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \\ &= \overline{Q}_k(\{x_k\}|x_{k-1})\overline{S}_k(\{o_k\}|x_k)\overline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k),\end{aligned}$$

using conjugacy and Equation (4), since $\underline{P}_{k+1}(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \leq 0$. This last inequality is true because we know that $\underline{P}_{k+1}(-\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) = -\overline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k)$ by conjugacy and that $\overline{P}_{k+1}(\mathbb{I}_{\{x_{k+1:n}\}}\mathbb{I}_{\{o_{k+1:n}\}}|x_k) \geq 0$ by Lemma 6.

Repeating the steps above again and again, eventually yields Equation (7):

$$\overline{P}_k(\mathbb{I}_{\{x_{k:n}\}}\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) = \prod_{i=k}^{n} \overline{Q}_i(\{x_i\}|x_{i-1})\overline{S}_i(\{o_i\}|x_i).$$

In the last step, for $k = n$, we have used the equality $\overline{E}_n(\{o_n\}|x_n) = \overline{S}_n(\{o_n\}|x_n)$, which follows from Equation (3) and conjugacy. □

**Lemma 6.** *Consider a coherent lower prevision $\underline{P}$ on $\mathscr{G}(\mathscr{X})$. Then, for all $f \in \mathscr{G}(\mathscr{X})$, we have that* $\min f \leq \underline{P}(f) \leq \overline{P}(f) \leq \max f$ *and, for all $\mu \in \mathbb{R}$, that $\underline{P}(f) = \overline{P}(\mu) = \mu$.*

*Proof.* We prove the inequalities in $\min f \leq \underline{P}(f) \leq \overline{P}(f) \leq \max f$ one by one. The first one is the same as C1. It follows by C3 that $\underline{P}(f - f) \geq \underline{P}(f) + \underline{P}(-f)$ and therefore, since we know by C2 that $\underline{P}(0) = 0\underline{P}(0) = 0$, this implies that $\underline{P}(f) \leq -\underline{P}(-f) = \overline{P}(f)$, using conjugacy for the last equality. For the gamble $-f$, C1 yields that $\min -f \leq \underline{P}(-f)$ which in turn implies that $\max f = -\min -f \geq -\underline{P}(-f) = \overline{P}(f)$.

To conclude, $\underline{P}(f) = \overline{P}(\mu) = \mu$ follows by applying these inequalities for $f = \mu$. □

*Proof of Proposition 1.* Observe that

$$\overline{P}_k(\mathbb{I}_{\{o_{k:n}\}}|x_{k-1}) = \overline{P}_k\left(\mathbb{I}_{\{o_{k:n}\}} \sum_{z_{k:n} \in \mathscr{X}_{k:n}} \mathbb{I}_{\{z_{k:n}\}}\middle|x_{k-1}\right) \geq \overline{P}_k\left(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{z^*_{k:n}\}}\middle|x_{k-1}\right) > 0,$$

where $z^*_{k:n}$ is any element of $\mathscr{X}_{k:n}$. The equality follows from $\sum_{z_{k:n} \in \mathscr{X}_{k:n}} \mathbb{I}_{\{z_{k:n}\}} = 1$, the first inequality from Lemma 8(ii), and the second one from the positivity assumption (10) and Equation (7).

In the same way, we can easily prove that

$$\overline{E}_k(\{o_{k:n}\}|x_k) = \overline{E}_k\left(\mathbb{I}_{\{o_{k:n}\}} \sum_{z_{k+1:n} \in \mathscr{X}_{k+1:n}} \mathbb{I}_{\{z_{k+1:n}\}}\middle|x_k\right) \geq \overline{E}_k\left(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{z^*_{k+1:n}\}}\middle|x_k\right) > 0.$$

This time, we have used the positivity assumption (10) and Equation (9) for the last inequality. □

*Proof of Theorem 2.* Consider the real-valued function $\rho$, defined by

$$\rho(\mu) := \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}} - \mu]) \text{ for all } \mu \in \mathbb{R}.$$

It follows from Equation (11) that $\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n})$ is $\rho$'s rightmost zero, and we also know that $\rho(0) = \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}])$. Furthermore, $\rho$ is non-increasing and continuous by Lemma 7(i), and has at least one zero by Lemma 7(ii). Hence, if $\rho(0) > 0$, then $\rho$ has at least one positive zero and $\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n}) > 0$. If $\rho(0) < 0$, then $\rho$ has only negative zeroes and we then find that $\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n}) < 0$. Hence, proving the theorem comes down to proving that $\rho(0) = 0$ implies that $\rho(\varepsilon) < 0$ for all $\varepsilon > 0$, since this in turn implies that $\underline{P}(\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}}|o_{1:n}) = 0$. We now prove this implication. We consider two different cases.

*The case* $x_1 = \hat{x}_1$. For any real $\varepsilon > 0$:

$$
\begin{aligned}
\rho(\varepsilon) &= \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}} - \varepsilon]) \\
&= \underline{Q}_1(\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}} - \varepsilon]|X_1)) \\
&= \underline{Q}_1\left(\mathbb{I}_{\{x_1\}}\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|x_1) + \sum_{z_1 \neq x_1} \mathbb{I}_{\{z_1\}}\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_1)\right).
\end{aligned} \tag{37}
$$

The coefficients $\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_1)$ can be written as $-\varepsilon\overline{E}_1(\{o_{1:n}\}|z_1)$ by conjugacy and C2, which makes them negative, decreasing functions of $\varepsilon$, since $\overline{E}_1(\{o_{1:n}\}|z_1) > 0$ by the positivity assumption (10) and Proposition 1.

For the coefficient $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|x_1)$, we consider two possible cases.

If $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}}]|x_1) > 0$, we know that $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|x_1)$ is a decreasing function of $\varepsilon$ by Lemma 7(vi). Therefore, the argument of $\underline{Q}_1$ in Equation (37) decreases pointwise in $\varepsilon$, which by Lemma 8(i) implies that $\rho(\varepsilon)$ is a decreasing function of $\varepsilon$ and therefore $\rho(\varepsilon) < \rho(0) = 0$.

If, on the other hand, $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}}]|x_1) \leq 0$, then we know by Lemma 8(ii) that $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|x_1) \leq 0$, implying that

$$
\begin{aligned}
\rho(\varepsilon) &\leq \underline{Q}_1\left(\sum_{z_1 \neq x_1} \mathbb{I}_{\{z_1\}}\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_1)\right) \\
&\leq \underline{Q}_1\left(\mathbb{I}_{\{z_{1*}\}}\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_{1*})\right) = -\varepsilon\overline{E}_1(\{o_{1:n}\}|z_{1*})\overline{Q}_1\{z_{1*}\} < 0.
\end{aligned}
$$

In this expression, $z_{1*}$ is an arbitrary $z_1 \neq x_1$. The first two inequalities are due to Lemma 8(ii). Conjugacy and C2 yield the equality and the last inequality is a consequence of the positivity assumption (10) and Proposition 1. Also in this case, therefore, we find that $\rho(\varepsilon) < 0$.

*The case* $x_1 \neq \hat{x}_1$. For any real $\varepsilon > 0$:

$$
\begin{aligned}
\rho(\varepsilon) &= \underline{P}(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}} - \varepsilon]) \\
&= \underline{Q}_1(\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{1:n}\}} - \mathbb{I}_{\{\hat{x}_{1:n}\}} - \varepsilon]|X_1)) \\
&= \underline{Q}_1\Bigg(\mathbb{I}_{\{x_1\}}\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \varepsilon]|x_1) + \mathbb{I}_{\{\hat{x}_1\}}\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[-\mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|\hat{x}_1) \\
&\qquad\qquad + \sum_{z_1 \neq x_1, \hat{x}_1} \mathbb{I}_{\{z_1\}}\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_1)\Bigg)
\end{aligned} \tag{38}
$$

In the proof for the case $x_1 = \hat{x}_1$, we have already shown that the coefficients $\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|z_1)$ are negative, decreasing functions of $\varepsilon$. Together with Lemma 8(ii), this allows us to infer that $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[-\mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|\hat{x}_1) \leq \underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|\hat{x}_1) < 0$, which in turn by Lemma 7(vii) implies that $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[-\mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|\hat{x}_1)$ is a decreasing function of $\varepsilon$. All that is left to consider is the coefficient $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \varepsilon]|x_1)$. There are two possibilities.

If $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}\mathbb{I}_{\{x_{2:n}\}}|x_1) > 0$, then Lemma 7(vi) implies that $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \varepsilon]|x_1)$ is a decreasing function of $\varepsilon$. Therefore, the argument of $\underline{Q}_1$ in Equation (38) decreases pointwise in $\varepsilon$, which by Lemma 8(i) implies that $\rho(\varepsilon)$ is a decreasing function of $\varepsilon$ and therefore $\rho(\varepsilon) < \rho(0) = 0$.

If, on the other hand, $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}\mathbb{I}_{\{x_{2:n}\}}|x_1) = 0$, then by Lemma 8(ii), $\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[\mathbb{I}_{\{x_{2:n}\}} - \varepsilon]|x_1) \leq 0$, implying that

$$\begin{aligned}\rho(\varepsilon) &\leq \underline{Q}_1(\mathbb{I}_{\{\hat{x}_1\}}\underline{E}_1(\mathbb{I}_{\{o_{1:n}\}}[-\mathbb{I}_{\{\hat{x}_{2:n}\}} - \varepsilon]|\hat{x}_1)) \\ &\leq \underline{Q}_1(\mathbb{I}_{\{\hat{x}_1\}}\underline{E}_1(-\varepsilon\mathbb{I}_{\{o_{1:n}\}}|\hat{x}_1)) = -\varepsilon\overline{E}_1(\{o_{1:n}\}|\hat{x}_1)\overline{Q}_1(\{\hat{x}_1\}) < 0.\end{aligned}$$

The first two inequalities follow from Lemma 8(ii). Conjugacy and C2 yield the equality, and the last inequality is a consequence of the positivity assumption (10) and Proposition 1. Also in this case, then, we find that $\rho(\varepsilon) < 0$. □

**Lemma 7.** *Let $\underline{P}$ be a coherent lower prevision on $\mathscr{G}(\mathscr{X})$. For any $f \in \mathscr{G}(\mathscr{X})$ and $y \in \mathscr{Y}$, consider the real-valued map $\rho$ defined on $\mathbb{R}$ by $\rho(\mu) := \underline{P}(\mathbb{I}_{\{y\}}[f - \mu])$ for all real $\mu$. Then the following statements hold:*

(i) *$\rho$ is non-increasing, concave and continuous.*

(ii) *$\rho$ has at least one zero.*

(iii) *If $\underline{P}(\{y\}) > 0$, then $\rho$ is decreasing and has a unique zero.*

(iv) *If $\overline{P}(\{y\}) = 0$, then $\rho$ is identically zero.*

(v) *If $\underline{P}(\{y\}) = 0$ and $\overline{P}(\{y\}) > 0$, then $\rho$ is zero on $(-\infty, \underline{P}(f|y)]$, and negative and decreasing on $(\underline{P}(f|y), +\infty)$.*

(vi) *If $\rho(a) > 0$ for some $a$, then $\rho$ is decreasing and has a unique zero.*

(vii) *If $\rho$ is negative on an interval $(a,b)$, then it is also decreasing on $(a,b)$.*

*Proof.* We start by proving (i). It follows directly from Lemma 8(ii) that $\rho$ is non-increasing in $\mu$. Now consider $\mu_1$ and $\mu_2$ in $\mathbb{R}$ and $0 \leq \lambda \leq 1$. $\rho$ is concave because

$$\begin{aligned}\rho(\lambda\mu_1 + (1-\lambda)\mu_2) &= \underline{P}(\mathbb{I}_{\{y\}}[f - (\lambda\mu_1 + (1-\lambda)\mu_2)]) \\ &= \underline{P}(\lambda\mathbb{I}_{\{y\}}[f - \mu_1] + (1-\lambda)\mathbb{I}_{\{y\}}[f - \mu_2]) \\ &\geq \underline{P}(\lambda\mathbb{I}_{\{y\}}[f - \mu_1]) + \underline{P}((1-\lambda)\mathbb{I}_{\{y\}}[f - \mu_2]) \\ &= \lambda\underline{P}(\mathbb{I}_{\{y\}}[f - \mu_1]) + (1-\lambda)\underline{P}(\mathbb{I}_{\{y\}}[f - \mu_2]) \\ &= \lambda\rho(\mu_1) + (1-\lambda)\rho(\mu_2),\end{aligned}$$

where the inequality follows from C3 and the subsequent step is due to C2. To prove that $\rho(\mu)$ is continuous, consider any $\mu_1$ and $\mu_2$ in $\mathbb{R}$, then we see that

$$\begin{aligned}\rho(\mu_2) &= \underline{P}(\mathbb{I}_{\{y\}}[f - \mu_2]) = \underline{P}(\mathbb{I}_{\{y\}}[f - \mu_1 + (\mu_1 - \mu_2)]) \\ &= \underline{P}(\mathbb{I}_{\{y\}}[f - \mu_1] + \mathbb{I}_{\{y\}}(\mu_1 - \mu_2)) \geq \underline{P}(\mathbb{I}_{\{y\}}[f - \mu_1]) + \underline{P}(\mathbb{I}_{\{y\}}(\mu_1 - \mu_2)) \\ &= \rho(\mu_1) - \overline{P}(\{y\}) \odot (\mu_2 - \mu_1),\end{aligned}$$

where the inequality follows from C3, and the last equality is due to conjugacy and C2. Hence $|\rho(\mu_1) - \rho(\mu_2)| \leq |\mu_2 - \mu_1|\overline{P}(\{y\})$, which proves that $\rho$ is Lipschitz continuous, and therefore also continuous.

To prove (ii), first notice that $\rho(\min f) = \underline{P}(\mathbb{I}_{\{y\}}[f-\min f]) \geq \underline{P}(\mathbb{I}_{\{y\}}[\min f - \min f]) = 0$ and $\rho(\max f) = \underline{P}(\mathbb{I}_{\{y\}}[f-\max f])E \leq \underline{P}(\mathbb{I}_{\{y\}}[\max f - \max f]) = 0$. The inequalities are a consequence of Lemma 8(ii), and the last equalities follow from Lemma 6. Since $\rho(\mu)$ is continuous, this implies the existence of a zero between $\min f$ and $\max f$.

Property (iii) can be proved by considering $\mu_1$ and $\mu_2$ in $\mathbb{R}$ with $\mu_2 > \mu_1$. If $\underline{P}(\{y\}) > 0$, we see that $\rho$ is decreasing, since

$$\begin{aligned}\rho(\mu_1) &= \underline{P}(\mathbb{I}_{\{y\}}[f-\mu_1]) = \underline{P}(\mathbb{I}_{\{y\}}[f-\mu_2+(\mu_2-\mu_1)]) \\ &= \underline{P}(\mathbb{I}_{\{y\}}[f-\mu_2]+\mathbb{I}_{\{y\}}(\mu_2-\mu_1)) \geq \underline{P}(\mathbb{I}_{\{y\}}[f-\mu_2])+\underline{P}(\mathbb{I}_{\{y\}}(\mu_2-\mu_1)) \\ &= \rho(\mu_2)+(\mu_2-\mu_1)\underline{P}(\{y\}) > \rho(\mu_2),\end{aligned}$$

where the first inequality follows from C3 and the last equality from C2. We know by (ii) that $\rho$ has at least one zero, which must be unique because $\rho$ is decreasing.

To prove (iv), first note that $\overline{P}(\{y\}) = 0$ also implies $\underline{P}(\{y\}) = 0$, because of Lemma 6. Now fix $\mu$ in $\mathbb{R}$ and choose $a$ and $b$ in $\mathbb{R}$ such that

$$a < \min\{0, \min\{f-\mu\}\} \leq \max\{0, \max\{f-\mu\}\} < b.$$

Then we find that $\rho(\mu) = \underline{P}(\mathbb{I}_{\{y\}}[f-\mu]) \geq \underline{P}(\mathbb{I}_{\{y\}}a) = a\overline{P}(\{y\}) = 0$ and $\rho(\mu) = \underline{P}(\mathbb{I}_{\{y\}}[f-\mu]) \leq \underline{P}(\mathbb{I}_{\{y\}}b) = b\underline{P}(\{y\}) = 0$, using Lemma 8(ii), C2 and conjugacy. We conclude that $\rho(\mu) = 0$ for any $\mu$ in $\mathbb{R}$.

The proof of (v) starts by noticing that $\rho(\mu) \geq 0$ for $\mu \in (-\infty, \underline{P}(f|y)]$ and $\rho(\mu) < 0$ for $\mu \in (\underline{P}(f|y), +\infty)$, due to the definition of $\underline{P}(f|y)$ (see Equation (11)), and the fact that $\rho$ is non-increasing by (i). In the proof of (iv), we have already shown that $\rho$ is non-positive if $\underline{P}(\{y\}) = 0$, which allows us to conclude that $\rho(\mu) = 0$ for $\mu \in (-\infty, \underline{P}(f|y)]$. We are left to prove that $\rho$ is decreasing on the interval $(\underline{P}(f|y), +\infty)$. We will do so by contradiction. Suppose that $\rho$ is not decreasing on that interval, then there are $\mu_1$ and $\mu_2$ in this interval, such that $\mu_2 > \mu_1$ and $0 > \rho(\mu_2) \geq \rho(\mu_1)$. Since $\rho$ is zero on $(-\infty, \underline{P}(f|y))$, we can also choose $\mu_0 < \mu_1$ such that $\rho(\mu_0) = 0$. The existence of such $\mu_0$, $\mu_1$ and $\mu_2$ contradicts the concavity of $\rho$, established by (i).

To prove (vi), observe that $\overline{P}(\{y\}) \geq \underline{P}(\{y\}) \geq 0$ by Lemma 6. This implies that the three cases considered in (iii), (iv) and (v) are exhaustive and mutually exclusive. If there is an $a$ for which $\rho(a) > 0$, we can only have the case considered in (iii), which implies that $\rho$ is decreasing and has a unique zero.

It now only remains to prove (vii). By repeating the argument in the proof of (vi), we see that $\rho$ is negative on an interval $(a,b)$, only the cases considered in (iii) and (v) can obtain. For (iii), $\rho$ is decreasing on its entire domain. For (v), $\rho$ is definitely decreasing on $(a,b)$. $\square$

**Lemma 8.** *Consider a coherent lower prevision $\underline{P}$ on $\mathscr{G}(\mathscr{X})$ and two gambles $f, g \in \mathscr{G}(\mathscr{X})$.*

(i) *If $f(x) > g(x)$ for all $x \in \mathscr{X}$, then $\underline{P}(f) > \underline{P}(g)$.*

(ii) *If $f(x) \geq g(x)$ for all $x \in \mathscr{X}$, then $\underline{P}(f) \geq \underline{P}(g)$.*

*Proof.* We start with (i). Since $f-g$ is pointwise positive, we have that $\min(f-g) > 0$ and therefore that $\underline{P}(f-g) \geq \min(f-g) > 0$, using C1 for the first inequality. It now follows from C3 that $\underline{P}(f) = \underline{P}((f-g)+g) \geq \underline{P}(f-g)+\underline{P}(g)$, and therefore that $\underline{P}(f)-\underline{P}(g) \geq \underline{P}(f-g) > 0$, whence indeed $\underline{P}(f) > \underline{P}(g)$. The proof for (ii) is analogous; this time, we have that $\min(f-g) \geq 0$ and therefore that $\underline{P}(f)-\underline{P}(g) \geq \underline{P}(f-g) \geq \min(f-g) \geq 0$. $\square$

*Proof of Equation* (15). Let $\Delta[x_{k:n},\hat{x}_{k:n}] := \mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]$. Since we are considering the case $k \in \{1,\ldots,n-1\}$ and $\hat{x}_k = x_k$, we find that

$$\begin{aligned}\Delta[x_{k:n},\hat{x}_{k:n}] = \mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}] &= \mathbb{I}_{\{o_k\}}\mathbb{I}_{\{x_k\}}\mathbb{I}_{\{o_{k+1:n}\}}[\mathbb{I}_{\{x_{k+1:n}\}} - \mathbb{I}_{\{\hat{x}_{k+1:n}\}}]\\ &= \mathbb{I}_{\{o_k\}}\mathbb{I}_{\{x_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}],\end{aligned}$$

which in turn implies that

$$\begin{aligned}\underline{P}_k(\Delta[x_{k:n},\hat{x}_{k:n}]|x_{k-1}) &= \underline{Q}_k(\underline{E}_k(\mathbb{I}_{\{o_k\}}\mathbb{I}_{\{x_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|X_k)|x_{k-1})\\ &= \underline{Q}_k(\mathbb{I}_{\{x_k\}}\underline{E}_k(\mathbb{I}_{\{o_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|x_k)|x_{k-1})\\ &= \overline{Q}_k(\{x_k\}|x_{k-1}) \odot \underline{E}_k(\mathbb{I}_{\{o_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|x_k)\\ &= \overline{Q}_k(\{x_k\}|x_{k-1})\overline{S}_k(\{o_k\}|x_k) \odot \underline{P}_{k+1}(\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|x_k),\end{aligned}$$

proving Equation (15). The first equality follows from Equation (5). The second equality holds because $\mathbb{I}_{\{x_k\}}(z_k) = 0$ for all $z_k \neq x_k$, implying that

$$\underline{E}_k(\mathbb{I}_{\{o_k\}}\mathbb{I}_{\{x_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|X_k) = \mathbb{I}_{\{x_k\}}\underline{E}_k(\mathbb{I}_{\{o_k\}}\Delta[x_{k+1:n},\hat{x}_{k+1:n}]|x_k).$$

The third equality is follows from conjugacy and C2, and the last one follows from Equation (4). □

*Proof of Equation* (16). Since $\hat{x}_n = x_n$, Lemma 6 yields:

$$\underline{P}_n(\mathbb{I}_{\{o_n\}}[\mathbb{I}_{\{x_n\}} - \mathbb{I}_{\{\hat{x}_n\}}|x_{n-1}) = \underline{P}_n(\mathbb{I}_{\{o_n\}}[\mathbb{I}_{\{x_n\}} - \mathbb{I}_{\{x_n\}}|x_{n-1}) = \underline{P}_n(0|x_{n-1}) = 0.$$ □

*Proof of Equation* (17). If $k \in \{1,\ldots,n\}$ and $\hat{x}_k \neq x_k$, then

$$\begin{aligned}\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}|x_{k-1}) &= \underline{Q}_k(\underline{E}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|X_k)|x_{k-1})\\ &= \underline{Q}_k(\mathbb{I}_{\{x_k\}}\underline{E}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{x_{k+1:n}\}}|x_k) + \mathbb{I}_{\{\hat{x}_k\}}\underline{E}_k(-\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{\hat{x}_{k+1:n}\}}|\hat{x}_k)|x_{k-1})\\ &= \underline{Q}_k(\mathbb{I}_{\{x_k\}}\underline{E}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{x_{k+1:n}\}}|x_k) - \mathbb{I}_{\{\hat{x}_k\}}\overline{E}_k(\mathbb{I}_{\{o_{k:n}\}}\mathbb{I}_{\{\hat{x}_{k+1:n}\}}|\hat{x}_k)|x_{k-1})\\ &= \underline{Q}_k(\mathbb{I}_{\{x_k\}}\beta(x_{k:n}) - \mathbb{I}_{\{\hat{x}_k\}}\alpha(\hat{x}_{k:n})|x_{k-1}),\end{aligned}$$

proving Equation (17). The reasons why all these equalities hold, are analogous to the ones given in the proof of Equation (15). □

*Proof of Theorem 3.* Fix $k \in \{1,\ldots,n-1\}$, $x_{k-1} \in \mathscr{X}_{k-1}$ and $\hat{x}_{k:n} \in \mathscr{X}_{k:n}$. We now assume that $\hat{x}_{k+1:n} \notin \operatorname{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n})$ and show that $\hat{x}_{k:n} \notin \operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. It follows from the assumption that $\underline{P}_{k+1}(\mathbb{I}_{\{o_{k+1:n}\}}[\mathbb{I}_{\{x_{k+1:n}\}} - \mathbb{I}_{\{\hat{x}_{k+1:n}\}}|\hat{x}_k) > 0$ for some $x_{k+1:n} \in \mathscr{X}_{k+1}$. Now prefix this state sequence $x_{k+1:n}$ with the state $\hat{x}_k$ to form the state sequence $x_{k:n}$, implying that $x_k = \hat{x}_k$. We then infer from Equation (15) that

$$\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}|x_{k-1}) = \underline{Q}_k(\{\hat{x}_k\}|x_{k-1})\underline{S}_k(\{o_k\}|\hat{x}_k)\underline{P}_{k+1}(\mathbb{I}_{\{o_{k+1:n}\}}[\mathbb{I}_{\{x_{k+1:n}\}} - \mathbb{I}_{\{\hat{x}_{k+1:n}\}}|\hat{x}_k) > 0,$$

which tells us that indeed $\hat{x}_{k:n} \notin \operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. □

*Proof of Equations* (33) *and* (34). First, we consider $k = n$. For every $x_{n-1} \in \mathscr{X}_{n-1}$, we determine $\operatorname{opt}(\mathscr{X}_n|x_{n-1}, o_n)$ as the set of those elements $\hat{x}_n$ of $\mathscr{X}_n$ for which

$$(\forall x_n \in \mathscr{X}_n \setminus \{\hat{x}_n\})\underline{Q}_n(\mathbb{I}_{\{x_n\}}\beta_n^{\max}(x_n) - \mathbb{I}_{\{\hat{x}_n\}}\alpha(\hat{x}_n)|x_{n-1}) \leq 0,$$

as this condition is equivalent to the optimality condition (14) for $k = n$, taking into account Equations (16), (17) and (31). We now show that this condition is also equivalent to

$$(\forall x_n \in \mathscr{X}_n \setminus \{\hat{x}_n\})\alpha(\hat{x}_n) \geq \beta_n^{\max}(x_n)\theta_n(\hat{x}_n, x_n|x_{n-1}), \tag{39}$$

To see this, we consider two different cases. For those $x_n$ for which $\beta_n^{\max}(x_n) = 0$, the inequalities $\underline{Q}_n(\mathbb{I}_{\{x_n\}}\beta_n^{\max}(x_n) - \mathbb{I}_{\{\hat{x}_n\}}\alpha(\hat{x}_n)|x_{n-1}) \leq 0$ and $\alpha(\hat{x}_n) \geq \beta_n^{\max}(x_n)\theta_n(\hat{x}_n, x_n|x_{n-1})$ are both trivially satisfied since $\alpha(\hat{x}_n) = \overline{S}_n(\{o_n\}|\hat{x}_n) > 0$ by the positivity assumption (10). If $\beta_n^{\max}(x_n) > 0$, both inequalities are equivalent because of C2 and Equation (27):

$$\begin{aligned}\underline{Q}_n(\mathbb{I}_{\{x_n\}}\beta_n^{\max}(x_n) - \mathbb{I}_{\{\hat{x}_n\}}\alpha(\hat{x}_n)|x_{n-1}) \leq 0 &\Leftrightarrow \underline{Q}_n\left(\mathbb{I}_{\{x_n\}} - \mathbb{I}_{\{\hat{x}_n\}}\frac{\alpha(\hat{x}_n)}{\beta_n^{\max}(x_n)}\middle|x_{n-1}\right) \leq 0 \\ &\Leftrightarrow \frac{\alpha(\hat{x}_n)}{\beta_n^{\max}(x_n)} \geq \theta_n(\hat{x}_n, x_n|x_{n-1}) \\ &\Leftrightarrow \alpha(\hat{x}_n) \geq \beta_n^{\max}(x_n)\theta_n(\hat{x}_n, x_n|x_{n-1}).\end{aligned}$$

Using Equation (32), Equation (39) can now be reformulated as $\alpha(\hat{x}_n) \geq \alpha_n^{\text{opt}}(\hat{x}_n|x_{n-1})$, which completes the proof of the equivalence.

Next, consider any $k \in \{1, \ldots, n-1\}$ and $x_{k-1} \in \mathscr{X}_{k-1}$. We must determine $\operatorname{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. We know from the Principle of Optimality (23) that we can limit the candidate optimal sequences $\hat{x}_{k:n}$ to the set $\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. Consider any such $\hat{x}_{k:n}$, then we must check for any $x_{k:n} \in \mathscr{X}_{k:n}$ whether $\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) \leq 0$; see Equation (14).

If $x_{k:n}$ is such that $x_k = \hat{x}_k$, this inequality is always satisfied. Indeed, if $\hat{x}_k \notin \operatorname{Pos}_k(x_{k-1})$, then we infer from Equation (25) that $\underline{Q}_k(\{\hat{x}_k\}|x_{k-1}) = 0$ or $\underline{S}_k(\{o_k\}|\hat{x}_k) = 0$, and then Equation (15) tells us that $\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) = 0$. If $\hat{x}_k \in \operatorname{Pos}_k(x_{k-1})$, we know from Equation (24) that $\hat{x}_{k+1:n} \in \operatorname{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n})$, which in turn implies that $\underline{P}_{k+1}(\mathbb{I}_{\{o_{k+1:n}\}}[\mathbb{I}_{\{x_{k+1:n}\}} - \mathbb{I}_{\{\hat{x}_{k+1:n}\}}]|\hat{x}_k) \leq 0$. Hence $\underline{P}_k(\mathbb{I}_{\{o_{k:n}\}}[\mathbb{I}_{\{x_{k:n}\}} - \mathbb{I}_{\{\hat{x}_{k:n}\}}]|x_{k-1}) \leq 0$, again by Equation (15).

This means we can limit ourselves to checking the inequality for those $x_{k:n}$ for which $x_k \neq \hat{x}_k$. So fix any $x_k \neq \hat{x}_k$, then we must check whether

$$(\forall x_{k+1:n} \in \mathscr{X}_{k+1:n})\underline{Q}_k(\mathbb{I}_{\{x_k\}}\beta(x_{k:n}) - \mathbb{I}_{\{\hat{x}_k\}}\alpha(\hat{x}_{k:n})|x_{k-1}) \leq 0;$$

see Equation (17). By Equation (28) and Lemma 8, this is equivalent to

$$\underline{Q}_k(\mathbb{I}_{\{x_k\}}\beta_k^{\max}(x_k) - \mathbb{I}_{\{\hat{x}_k\}}\alpha(\hat{x}_{k:n})|x_{k-1}) \leq 0,$$

which can in turn be seen to be equivalent to $\alpha(\hat{x}_{k:n}) \geq \beta_k^{\max}(x_k)\theta_k(\hat{x}_k, x_k|x_{k-1})$, using a course of reasoning completely analogous to the one used above for the case $k = n$. Since this inequality must hold for every $x_k \neq \hat{x}_k$, we infer from Equation (32) that we must have that $\alpha(\hat{x}_{k:n}) \geq \alpha_k^{\text{opt}}(\hat{x}_k|x_{k-1})$. So we must check this condition for all the candidate sequences $\hat{x}_{k:n}$ in $\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$, which proves Equation (33). $\square$

*Proof of Theorem 4.* We start by proving that every sequence $\hat{x}_{k:n}$ that is added in Line 2 of the Procedure `Recur`$(\hat{x}_{k:n}, n)$ is indeed an element of $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. If Line 2 of the Procedure `Recur`$(\hat{x}_{k:n}, n)$ is executed, this means that the Procedure `Recur`$(\hat{x}_{k:n-1}, n-1)$ was executed in the previous step, and that at that point, the if-conditions in Lines 5 and 6 were satisfied. Due to the first if-condition, we know that $\hat{x}_{k:n} \in \mathrm{cand}_{\hat{x}_{k:n}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ and therefore, by Equation (35), also that $\hat{x}_{k:n} \in \mathrm{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. From the second if-condition, we infer that $\alpha_n^{\max}(\hat{x}_n) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_{k:n}|x_{k-1})$, which can be seen to be equivalent with $\alpha(\hat{x}_{k:n}) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$, by Equation (31) and the repeated use of Equations (36) and (20). It now follows from Equation (33) that $\hat{x}_{k:n}$ is an element of $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$.

To conclude the proof, we show that a sequence $\hat{x}_{k:n}$ that has not been added during the course of the algorithm cannot be an element of $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. If a sequence $\hat{x}_{k:n}$ has not been added, this either implies that it is not an element of $\mathrm{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ [the if-condition on Line 5 of the Procedure `Recur` was not satisfied], or that there is some $i \in \{k, \ldots, n\}$ for which $\alpha_i^{\max}(\hat{x}_i) < \alpha_k^{\mathrm{opt}}(\hat{x}_{k:i}|x_{k-1})$ [the if-condition on Line 9' of Algorithm 2 or Line 5 of the Procedure `Recur` was not satisfied]. In the first case, it follows directly from Equation (33) that $\hat{x}_{k:n}$ cannot be an element of $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. In the second case, we find that $\alpha_i^{\max}(\hat{x}_i) < \alpha_k^{\mathrm{opt}}(\hat{x}_{k:i}|x_{k-1})$ implies that $\alpha(\hat{x}_{k:n}) < \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$, which can be seen to be equivalent with $\alpha(\hat{x}_{k:n}) < \alpha_k^{\mathrm{opt}}(\hat{x}_k|x_{k-1})$ by the repeated use of Equations (36) and (20). It then follows from Equation (33) that $\hat{x}_{k:n}$ cannot be an element of $\mathrm{opt}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. $\square$

*Proof of Theorem 5.* Equation (28) implies that there is at least one sequence $x_{s+1:n}^* \in \mathscr{X}_{s+1:n}$ for which $\alpha(\hat{x}_s \oplus x_{s+1:n}^*) = \alpha_s^{\max}(\hat{x}_s)$. We prove that the first state $x_{s+1}^*$ of this sequence meets both criteria of the theorem.

We know that $\mathrm{cand}_{\hat{x}_{k:s}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ and $\alpha_s^{\max}(\hat{x}_s) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_{k:s}|x_{k-1})$ because both conditions are necessary in order for the Procedure `Recur`$(\hat{x}_{k:s}, s)$ to be executed while running Algorithm 2. For $s = k$, the condition $\mathrm{cand}_{\hat{x}_{k:s}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ is not explicitely checked by Algorithm 2, but nevertheless also true because of Equations (24) and (35) and because we know that $\mathrm{opt}(\mathscr{X}_{k+1:n}|\hat{x}_k, o_{k+1:n}) \neq \emptyset$ [because every finite partially ordered set has at least one maximal element].

Since $\alpha(\hat{x}_s \oplus x_{s+1:n}^*) = \alpha_s^{\max}(\hat{x}_s) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_{k:s}|x_{k-1})$, we know from Equations (20) and (36) that $\alpha(x_{s+1:n}^*) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_{k:s} \oplus x_{s+1}^*|x_{k-1})$. By combining this with Equation (28), we find that $\alpha_{s+1}^{\max}(x_{s+1}^*) \geq \alpha_k^{\mathrm{opt}}(\hat{x}_{k:s} \oplus x_{s+1}^*|x_{k-1})$, meaning that $x_{s+1}^*$ satisfies the if-condition in Line 6.

Due to Lemma 9, we can infer from $\mathrm{cand}_{\hat{x}_{k:s}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ and $\alpha(\hat{x}_s \oplus x_{s+1:n}^*) = \alpha_s^{\max}(\hat{x}_s)$ that $\mathrm{cand}_{\hat{x}_{k:s} \oplus x_{s+1}^*}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$, meaning that $x_{s+1}^*$ satisfies the if-condition in Line 5 as well. $\square$

**Lemma 9.** *Consider any $k \in \{1, \ldots, n-1\}$, $s \in \{k, \ldots, n-1\}$, $x_{k-1} \in \mathscr{X}_{k-1}$, $x_{k:s} \in \mathscr{X}_{k:s}$ and $x_{s+1:n}^* \in \mathscr{X}_{s+1:n}$. Then if $\mathrm{cand}_{x_{k:s}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$ and $\alpha(x_s \oplus x_{s+1:n}^*) = \alpha_s^{\max}(x_s)$, we also have that $\mathrm{cand}_{x_{k:s} \oplus x_{s+1}^*}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$.*

*Proof.* Let $z_{s+1:n}$ be any sequence in $\mathscr{X}_{s+1:n}$ for which $x_{k:s} \oplus z_{s+1:n} \in \mathrm{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$; this is possible because, by assumption, $\mathrm{cand}_{x_{k:s}}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n}) \neq \emptyset$.

If there is some $q \in \{k, \ldots, s-1\}$ for which $x_q \notin \mathrm{Pos}_q(x_{q-1})$, then we denote the smallest such $q$ as $q^*$. In that case, by Equation (24), we find that $x_{q^*:s} \oplus x_{s+1:n}^*$ and $x_{q^*:s} \oplus z_{s+1:n}$ are both elements of $\mathrm{cand}(\mathscr{X}_{q^*:n}|x_{q^*-1}, o_{q^*:n})$. If no such $q$ exists, we let $q^* := s$. In that case, since $x_q \in \mathrm{Pos}_q(x_{q-1})$ for all

$q \in \{k, \ldots, s-1\}$, it follows from $x_{k:s} \oplus z_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ and the repeated use of Equations (24) and (23) that $x_s \oplus z_{s+1:n}$ belongs to $\operatorname{cand}(\mathscr{X}_{s:n}|x_{s-1}, o_{s:n})$. Since $\alpha(x_s \oplus x^*_{s+1:n}) = \alpha^{\max}_s(x_s)$, we can infer from Lemma 10 that $x^*_{s+1:n} \in \operatorname{opt}(\mathscr{X}_{s+1:n}|x_s, o_{s+1:n})$ and therefore, by Equation (24), that $x_s \oplus x^*_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{s:n}|x_{s-1}, o_{s:n})$.

In any case, we now have a $q^* \in \{k, \ldots, s\}$ for which $x_{q^*:s} \oplus x^*_{s+1:n}$ and $x_{q^*:s} \oplus z_{s+1:n}$ belong to $\operatorname{cand}(\mathscr{X}_{q^*:n}|x_{q^*-1}, o_{q^*:n})$ and for which, for all $q \in \{k, \ldots, q^*-1\}$, $x_q \in \operatorname{Pos}_q(x_{q-1})$. If $q^* = k$, this concludes the proof. Therefore, we will from now on consider the case $q^* \in \{k+1, \ldots, s\}$.

We first recall that $\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ can be constructed by applying Equations (33) and (24) repeatedly. Therefore, since we know that $x_{q^*:s} \oplus z_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{q^*:n}|x_{q^*-1}, o_{q^*:n})$ and $x_{k:s} \oplus z_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$, it must be that

$$\alpha(x_{q:s} \oplus z_{s+1:n}) \geq \alpha^{\mathrm{opt}}_q(x_q|x_{q-1}) \text{ for all } q \in \{k+1, \ldots, q^*\}. \tag{40}$$

Furthermore, since $\alpha(x_s \oplus x^*_{s+1:n}) = \alpha^{\max}_s(x_s)$, we infer from Equation (28) that $\alpha(x_s \oplus x^*_{s+1:n}) \geq \alpha(x_s \oplus z_{s+1:n})$ and therefore, by Equation (20), we find hat

$$\alpha(x_{q:s} \oplus x^*_{s+1:n}) \geq \alpha(x_{q:s} \oplus z_{s+1:n}) \text{ for all } q \in \{k+1, \ldots, s\}.$$

Hence, by Equation (40):

$$\alpha(x_{q:s} \oplus x^*_{s+1:n}) \geq \alpha^{\mathrm{opt}}_q(x_q|x_{q-1}) \text{ for all } q \in \{k+1, \ldots, q^*\}. \tag{41}$$

Since $\operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$ can be constructed by repeatedly applying Equations (33) and (24) and because $x_{q^*:s} \oplus x^*_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{q^*:n}|x_{q^*-1}, o_{q^*:n})$, we now infer from Equation (41) that $x_{k:s} \oplus x^*_{s+1:n} \in \operatorname{cand}(\mathscr{X}_{k:n}|x_{k-1}, o_{k:n})$. $\square$

**Lemma 10.** *Consider any $s \in \{1, \ldots, n-1\}$, $x_s \in \mathscr{X}_s$ and $x^*_{s+1:n} \in \mathscr{X}_{s+1:n}$. Then*

$$\alpha(x_s \oplus x^*_{s+1:n}) = \alpha^{\max}_s(x_s) \implies x^*_{s+1:n} \in \operatorname{opt}(\mathscr{X}_{s+1:n}|x_s, o_{s+1:n}).$$

*Proof.* Assume that $\alpha(x_s \oplus x^*_{s+1:n}) = \alpha^{\max}_s(x_s)$ and consider any $z_{s+1:n} \in \mathscr{X}_{s+1:n}$. Then we know by Equation (28) that $\alpha(x_s \oplus x^*_{s+1:n}) \geq \alpha(x_s \oplus z_{s+1:n})$ and therefore, by Equation (19) and (7), that

$$\overline{S}_s(\{o_s\}|x_s)\overline{P}_{s+1}(\mathbb{I}_{\{x^*_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s) \geq \overline{S}_s(\{o_s\}|x_s)\overline{P}_{s+1}(\mathbb{I}_{\{z_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s).$$

Together with the positivity assumption (10), this implies that

$$\overline{P}_{s+1}(\mathbb{I}_{\{x^*_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s) \geq \overline{P}_{s+1}(\mathbb{I}_{\{z_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s). \tag{42}$$

By C3, we also know that

$$\underline{P}_{s+1}(-\mathbb{I}_{\{x^*_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s) \geq \underline{P}_{s+1}(\mathbb{I}_{\{o_{s+1:n}\}}(\mathbb{I}_{\{z_{s+1:n}\}} - \mathbb{I}_{\{x^*_{s+1:n}\}})|x_s) + \underline{P}_{s+1}(-\mathbb{I}_{\{z_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s)$$

which, by conjugacy, implies that

$$\underline{P}_{s+1}(\mathbb{I}_{\{o_{s+1:n}\}}(\mathbb{I}_{\{z_{s+1:n}\}} - \mathbb{I}_{\{x^*_{s+1:n}\}})|x_s) \leq \overline{P}_{s+1}(\mathbb{I}_{\{z_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s) - \overline{P}_{s+1}(\mathbb{I}_{\{x^*_{s+1:n}\}}\mathbb{I}_{\{o_{s+1:n}\}}|x_s).$$

Using Equation (42), we see that $\underline{P}_{s+1}(\mathbb{I}_{\{o_{s+1:n}\}}(\mathbb{I}_{\{z_{s+1:n}\}} - \mathbb{I}_{\{x^*_{s+1:n}\}})|x_s) \leq 0$. Since this holds for all $z_{s+1:n} \in \mathscr{X}_{s+1:n}$, we infer from Equation (14) that $x^*_{s+1:n} \in \operatorname{opt}(\mathscr{X}_{s+1:n}|x_s, o_{s+1:n})$. $\square$

## References


Bellman, R. (1957). *Dynamic Programming*. Princeton University Press, Princeton.

Benavoli, A., Zaffalon, M., & Miranda, E. (2011). Robust filtering through coherent lower previsions. *Automatic Control, IEEE Transactions on*, *56*(7), 1567–1581.

Brown, D. G., & Golod, D. (2010). Decoding HMMs using the k best paths: algorithms and applications.. *BMC Bioinformatics*, *11*(S-1), 28.

Cozman, F. G. (2000). Credal networks. *Artificial Intelligence*, *120*, 199–233.

Cozman, F. G. (2005). Graphical models for imprecise probabilities. *International Journal of Approximate Reasoning*, *39*(2-3), 167–184.

de Campos, L. M., Huete, J. F., & Moral, S. (1994). Probability intervals: a tool for uncertain reasoning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *2*, 167–196.

De Cooman, G., Miranda, E., & Zaffalon, M. (2011). Independent natural extension. *Artificial Intelligence*, *175*, 1911–1950.

De Cooman, G., Hermans, F., Antonucci, A., & Zaffalon, M. (2010). Epistemic irrelevance in credal nets: the case of imprecise Markov trees. *International Journal of Approximate Reasoning*, *51*, 1029–1052.

De Cooman, G., & Troffaes, M. C. M. (2005). Dynamic programming for deterministic discrete-time systems with uncertain gain. *International Journal of Approximate Reasoning*, *39*, 257–278.

De Cooman, G., Troffaes, M. C. M., & Miranda, E. (2008). *n*-Monotone exact functionals. *Journal of Mathematical Analysis and Applications*, *347*, 143–156.

Dempster, A. P. (1967). Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematical Statistics*, *38*, 325–339.

Huntley, N., & Troffaes, M. C. M. (2010). Normal form backward induction for decision trees with coherent lower previsions. *Annals of Operations Research*. Submitted for publication.

Kikuti, D., Cozman, F., & de Campos, C. (2005). Partially ordered preferences in decision trees: computing strategies with imprecision in probabilities. In *IJCAI Workshop About Advances on Preference Handling*, pp. 1313–1318.

Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press.

Mauá, D., de Campos, C., Benavoli, A., & Antonucci, A. (2013). On the complexity of strong and epistemic credal networks. In *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence*, pp. 391–400. AUAI Press.

Miranda, E. (2008). A survey of the theory of coherent lower previsions. *International Journal of Approximate Reasoning*, *48*(2), 628–658.

Miranda, E. (2009). Updating coherent lower previsions on finite spaces. *Fuzzy Sets and Systems*, *160*(9), 1286–1307.

Miranda, E., & de Cooman, G. (2007). Marginal extension in the theory of coherent lower previsions. *International Journal of Approximate Reasoning*, *46*(1), 188–225.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Shafer, G. (1976). *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, NJ.

Troffaes, M. C. M. (2007). Decision making under uncertainty using imprecise probabilities. *International Journal of Approximate Reasoning*, *45*(1), 17–29.

Utkin, L. V., & Augustin, T. (2005). Powerful algorithms for decision making under partial prior information and general ambiguity attitudes. In *in: Proceedings of the 3th International Symposium on Imprecise Probability: Theories and Applications (ISIPTA), Prague,Czech Republic*, pp. 349–358.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, *13*(2), 260–269.

Walley, P. (1991). *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London.

Walley, P. (1996). Inferences from multinomial data: learning about a bag of marbles. *Journal of the Royal Statistical Society, Series B*, *58*, 3–57. With discussion.

Weichselberger, K. (2000). The theory of interval probability as a unifying concept for uncertainty. *International Journal of Approximate Reasoning*, *24*(2–3), 149–170.