International Journal of Approximate Reasoning 95 (2018) 1–21

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Uncertain Logic Processing: logic-based inference and reasoning using Dempster–Shafer models

Rafael C. Núñez [a,*], Manohar N. Murthi [a], Kamal Premaratne [a], Matthias Scheutz [b], Otávio Bueno [c]

[a] *Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL, USA*
[b] *Department of Computer Science, Tufts University, Medford, MA, USA*
[c] *Department of Philosophy, University of Miami, Coral Gables, FL, USA*

ARTICLE INFO

Article history:
Received 23 January 2017
Received in revised form 7 November 2017
Accepted 29 December 2017
Available online 3 February 2018

Keywords:
Uncertain logic
Uncertain logic processing
First order logic
Automated reasoning
Dempster–Shafer theory
Belief theory

ABSTRACT

First order logic lies at the core of many methods in mathematics, philosophy, linguistics, and computer science. Although important efforts have been made to extend first order logic to the task of handling uncertainty, existing solutions are sometimes limited by the way they model uncertainty, or simply by the complexity of the problem formulation. These approaches could be strengthened by adding more flexibility in assigning probabilities (e.g., through intervals) and a more rigorous method of assigning probability/uncertainty measures. In this paper we present the basic theory of *Uncertain Logic Processing* (ULP), a robust framework for modeling and inference when information is available in the form of first order logic formulas subject to uncertainty. Dempster–Shafer (DS) theory provides the substrate for uncertainty modeling in the proposed ULP formulation. ULP can be tuned to preserve consistency with classical logic, allowing it to incorporate typical inference rules and properties, while preserving the strength of DS theory for representing and manipulating uncertainty.

1. Introduction

Due to its richness and versatility for representing and dealing with knowledge, First Order Logic (FOL) is regarded as one of the preferred knowledge representation languages for automated reasoning and inference systems. However, an exploding amount of typically uncertain data, information sources, and conflicting evidence keeps demanding improvements to FOL representations to enhance its capabilities, robustness, and efficiency.

In an effort to enrich FOL for reasoning in the presence of uncertainty, several approaches have focused on extending it using probabilistic measures. Examples of these approaches span from the already mature probabilistic logic [1] and fuzzy logic [2] theories, to the more contemporary Markov Logic Networks (MLNs) [3] and Probabilistic Soft Logic (PSL) [4] frameworks. By relying on graphical model structures, the latter approaches (i.e., MLNs and PSL) are perhaps better equipped for reasoning with large amounts of data. These approaches, however, may be hampered by scarce training data, conflicting evidence, or when the uncertainty is characterized as intervals. Our goal is to provide a framework that overcomes these issues, and complements probability-based logic reasoning systems with the ability of representing uncertainty using intervals. We call this framework Uncertain Logic Processing (ULP).

* Corresponding author.
E-mail address: nunez@umiami.edu (R.C. Núñez).

https://doi.org/10.1016/j.ijar.2017.12.006

At its core, ULP extends the scope of FOL by adding uncertainty management via Dempster–Shafer (DS) models. By relying on DS theory, ULP inherits the robustness for dealing with conflicting evidence [5] and provides a more flexible alternative for representing uncertainty through intervals. With a solid theoretical foundation, ULP is flexible enough to support reasoning systems that follow a variety of logic-based inference techniques (e.g., logic inference and satisfiability for both classical logic and paraconsistent logic models), and provides a substrate for creating efficient automated reasoning frameworks through optimization formulations or graphical model approaches.

1.1. Related research

When addressing quantification and propagation of uncertainty in logic reasoning systems, one of the earliest approaches is probabilistic logic [1]. Probabilistic logic provides a generalization of logic in which the truth values of sentences are probability values (between 0 and 1). A related approach, possibilistic logic [6], defines mechanisms (based on possibility theory) to associate logic formulas with weights. To efficiently address the computational complexity of larger problems, two new probability-enhanced methods have been recently introduced, namely, MLNs [3] and PSL [4]. MLNs combine FOL and Markov networks as undirected graphical models and assign weights to FOL formulas whose truth values are represented through probability values. PSL also uses undirected graphical models to represent templates for FOL formulas, and represents truth values as a number in the interval $[0, 1]$. PSL relies on the Lukasiewicz t-norm and its corresponding co-norm to model AND and OR operations. Both MLN and PSL are statistical methods that attempt to find uncertainty parameters that ensure satisfiability of the logical models specified by the users. Their definition, however, does not insist on being consistent with classical logic, as is our goal with ULP, and their accuracy may be compromised when the amount of training data is small.

Other approaches that extend logic reasoning to address uncertain scenarios are many-valued logics and fuzzy logic. Many-valued logics [7] do not restrict the number of truth values of propositions to two. Fuzzy logic is based on the theory of fuzzy sets [2]. In fuzzy logic, the imprecision in probabilities is modeled through membership functions defined on the sets of possible probabilities and utilities. Although useful in some applications, these logic reasoning systems introduce new design problems: What is the appropriate number of truth levels to be used in an application? How should we define the membership functions? As an alternative, in ULP we extend logic reasoning systems with the capability of modeling uncertainties based on intervals, without attaching new design requirements such as the definition of membership functions.

Regarding the use of intervals as a means of representing uncertainty, it appears in several methods, such as possibility theory [8] and DS theory. The latter, incorporates a rigorous methodology for assigning probabilistic measures based on available evidence [9]. Given the direct relation that exists between DS theory and probability (DS belief and plausibility measures are closely related to lower and upper probabilities [9]), it is possible to simplify DS models to probabilistic models. Considering these advantages, a number of researchers have studied the relation of DS theory and logic. The work in [10] is similar to the probabilistic logic in [1], but uses epistemic logics and interval-based extensions of probability functions (e.g., lower and upper probabilities, DS belief and plausibility bounds) instead of probabilities of possible worlds. In [11], DS theory is formulated in terms of propositional logic, enabling certain logic reasoning operations in the DS framework. Further work in [12] integrates this logic formulation of DS theory in a system for visual recognition. Motivated by the need for enhancing expressiveness in DS models, [13] proposes a more general formulation of DS models as propositional logic operations than [11]. A belief-function logic that uses DS models and operations to quantify and estimate uncertainty of logic formulas is introduced in [14]. This logic system allows non-zero belief assignments to the empty set and relies on Dempster's combination rule as the method for quantifying the propagation of uncertainty. It is also used in deduction systems where the logic formulas are in Skolemized normal conjunctive form. An application of this system for inference is described in [15]. Another application of DS-based extensions of logic systems is in [16], where information coming from sensors is modeled in DS theory, and then input into AgentSpeak [17], a logic-based inference system. Further analysis on DS-based logic is presented in [18]. A detailed study on uncertain implication rules is in [19]. This latter work, however, is focused on modeling causal probabilistic relations, and does not provide consistency with classical logic, as ULP does. In general, existing work on combining DS theory and logic does not necessarily aim to attain consistency with classical logic, as is our goal with the ULP, and are still limited to propositional logic models.

1.2. Contributions

In this paper we formalize the fundamental theory of ULP. ULP is an extension of FOL into the DS theoretic framework. In ULP, the uncertainty of logic expressions such as $\varphi(x)$ is quantified using an uncertainty interval $[\alpha, \beta]$, $0 \leq \alpha \leq \beta \leq 1$. The ULP framework allows us to model the uncertainty of this expression, and to combine it with similar expressions in order to solve various inference and reasoning problems. When $\alpha = \beta$, ULP renders probabilistic results. When $\alpha = \beta \in \{0, 1\}$, ULP converges to FOL. ULP can be used as an adaptive many-valued logic, with the quantization of the truth space varying according to the granularity defined by the input data. ULP can preserve consistency with classical logic. By preserving this consistency, it is possible to seamlessly move between the classical logic and DS domains, and to incorporate both the strength of FOL for information representation and inference, and the strength of DS for representing and manipulating uncertainty.

This paper extends original preliminary work that was outlined in [20,21]. In particular, this paper includes revised ULP definitions and notation, a more thorough description of the satisfiability problem in ULP, and a new example that addresses a bigger testing scenario.

The remainder of the paper is as follows:

- Sections 2 and 3 provide basic definitions of the DS theoretical framework and the logic models used in ULP, respectively;
- Section 4 provides definitions of basic ULP operators (e.g., NOT, AND, OR), based on generic DS fusion operators;
- Section 5 provides guidelines for the selection of appropriate DS fusion operators, as needed for enforcing particular properties of ULP operators; moreover, it defines a DS fusion operator for classically-consistent ULP;
- Section 6 introduces quantifiers for ULP;
- Section 7 provides guidelines for inference using ULP, and illustrates the process with an example;
- Section 8 introduces a method for scalable reasoning in ULP based on satisfiability formulations, and illustrates the method with examples;
- Finally, Section 9 concludes the paper.

2. Preliminaries: Dempster–Shafer theory

DS Theory is defined for a discrete set of events related to a given problem. This set is called the *Frame of Discernment* (FoD). For our purposes, we take the FoD to be defined on the finite set $\Theta = \{\theta_1, \theta_2, \ldots, \theta_N\}$, N being the cardinality $|\Theta|$ of the FoD. Elements $\theta_i \in \Theta$ represent the lowest level of discernible information. The power set of Θ is defined as a set containing all the possible subsets of Θ. The cardinality of the power set of Θ is 2^N. Next we introduce some basic definitions of DS theory. For additional details on DS theory, we refer the reader to [22,23].

Basic belief assignment and focal sets. A Basic Belief Assignment (BBA) or *mass assignment* is a mapping $m_\Theta(\cdot) : 2^\Theta \to [0,1]$ such that: $\sum_{A\subseteq\Theta} m_\Theta(A) = 1$ and $m_\Theta(\emptyset) = 0$. The BBA measures the support assigned to $A \subseteq \Theta$.

The subsets $A \subseteq \Theta$ such that $m(A) > 0$ are the focal elements of the BBA. The set of focal elements is the core $\mathcal{F}$. The triple $\{\Theta, \mathcal{F}, m_\Theta\}$ is called the *Body of Evidence* (BoE).

The state of complete ignorance is captured via $m(A) = 1_\Theta \equiv 1$ if $A = \Theta$; 0 if $A \subset \Theta$.

Belief, plausibility, and uncertainty. The belief and plausibility functions are associated with a BBA m. When focal elements are constituted of singletons only, the BBA, belief, and plausibility, all reduce to a probability assignment.

Given a BoE $\{\Theta, \mathcal{F}, m\}$, the *belief* function $\mathrm{Bel} : 2^\Theta \to [0,1]$ is defined as: $\mathrm{Bel}_\Theta(A) = \sum_{B\subseteq A} m_\Theta(B)$. $\mathrm{Bel}(A)$ represents the total belief that is committed to A without also being committed to its complement A^C. The *plausibility* function $\mathrm{Pl} : 2^\Theta \to [0,1]$ is defined as: $\mathrm{Pl}_\Theta(A) = 1 - \mathrm{Bel}_\Theta(A^C)$, where A^C denotes the complement of A in Θ, i.e., $A^C = \Theta \setminus A$. It corresponds to the total belief that does not contradict A. The *uncertainty* of A is captured by the interval: $\mathrm{Un}_A = [\mathrm{Bel}_\Theta(A), \mathrm{Pl}_\Theta(A)]$.

Combination rules. Information from distinct sources can be fused using combination (or fusion) rules. Perhaps the most widely used rule is *Dempster's Combination Rule (DCR)*. For two focal sets $C \subseteq \Theta$ and $D \subseteq \Theta$ such that $B = C \cap D$, and the two BBAs $m_j(\cdot)$ and $m_k(\cdot)$, the fused mass BBA $m_{jk}(B)$ generated by the DCR is given by:

$$m_{jk}(B) = \frac{1}{1 - K_{jk}} \sum_{C\cap D=B;\, B\neq\emptyset} m_j(C)\, m_k(D), \tag{1}$$

where $K_{jk} = \sum_{C\cap D=\emptyset} m_j(C)\, m_k(D) \neq 1$ is referred to as the *conflict* between the two BBAs.

The DCR is not defined for conflicting BBAs (i.e., when $K_{jk} = 1$). As an alternative, a combination rule that is robust in the presence of conflicting evidence is the *Conditional Fusion Equation (CFE)* in [24] which is based on the *Conditional Update Equation (CUE)* in [25]. For M BoEs, each possessing an identical FoD Θ, CFE-based fusion is defined by [24]

$$m(B) = \sum_{i=1}^{M} \sum_{A_i\in\mathcal{F}_i} \gamma_i(A_i)\, m_i(B|A_i), \tag{2}$$

where γ_i are non-negative real parameters which satisfy $\sum_{i=1}^{M} \sum_{A_i\in\mathcal{F}_i} \gamma_i(A_i) = 1$. The conditional masses above are computed using Fagin–Halperns' Rule of Conditioning [26]. For CFE-based fusion, the use of Fagin–Halperns' conditional is preferred over other DS theoretic conditionals because it provides a more natural and fluid transition to Bayesian models [27,25].

As will be explained later, in addition to DCR and CFE, ULP definitions described in this paper can be easily extended to incorporate other combination rules, as needed for the application at hand.

3. From propositional logic to uncertain logic processing: basic definitions of ULP expressions

3.1. Propositional logic

To introduce terminology and notation employed in our presentation of ULP, let us first recall that, in propositional logic, a *proposition* or a *sentence* is a statement that could take a truth-value. "The apple is on the table" or "if it rains then I get wet" are examples of propositions. Propositions are typically represented by lower case Greek letters (e.g., φ, ψ).

A proposition can be combined with other propositions using connectives. In this paper we only consider the following connectives: $\wedge$ (AND), $\vee$ (OR), $\neg$ (NOR), and $\Longrightarrow$ (implies).

Through *inference*, a group of propositions (called *premises*) are used to derive conclusions. Inference is typically a multi-step process, where each step must be sanctioned by an acceptable rule of inference.

In general, when referring to propositional logic, we follow the conventions and definitions provided in [28] and [29].

3.2. First-Order Logic (FOL)

FOL extends the scope of propositional logic and provides a more expressive framework for logic. FOL allows one to work with objects, relations, and functions. These can be expressed through *predicates*. Predicates could be described as strings of characters arranged according to rules of grammar. For example, if we want to express the relation *a* is above *b*, we can use $\texttt{Above}(\texttt{a},\texttt{b})$ [28]. We could also express a predicate using symbols. For example, in the expression above, we can make $\varphi =$ "Above" and obtain $\varphi(a,b)$. Note that, in this example, we are allowing φ to have arguments. These arguments could be either *constants* or *variables*.

FOL also introduces the use of quantifiers. Quantified sentences provide a more flexible way of talking about objects. There are two types of quantifiers: the *universal* ($\forall$) and the *existential* ($\exists$) quantifiers. A universally quantified sentence is formed by combining the universal quantifier $\forall$, a variable x, and a sentence φ, as follows: $\forall x\ \varphi(x)$. The intended meaning is that the sentence φ is true, no matter what object the variable x represents. An existentially quantified sentence is formed by combining the existential quantifier $\exists$, a variable x, and a sentence φ as follows: $\exists x\ \varphi(x)$. The intended meaning is that the sentence φ is true, for at least one object in the universe of discourse.

3.3. Uncertain Logic Processing (ULP)

ULP is an extension of FOL that allows one to deal with expressions whose truth is uncertain. The level of uncertainty is modeled with DS theory, and is bounded in the range $[0,1]$. In our presentation of ULP, we define the basic operators and symbols enumerated for propositional logic above (i.e., $\neg, \wedge, \vee$, and $\Longrightarrow$), and incorporate the symbols and quantifiers defined for FOL in Section 3.2, namely $\forall$ and $\exists$.

As an extension of FOL, we must also define the set of objects that ULP deals with, i.e., its domain. Furthermore, we need means for specifying exactly which objects are referred to by constants and variables in ULP expressions. For the purpose of the description herein, these basic components are defined as follows.

Definition 1 *(Domain and interpretation of a generic ULP model).* Let $D = \{d_1, d_2, \ldots, d_n\}$ be a non-empty set of individuals. Then, we define an interpretation function I as a function that maps an arbitrary variable x into an element $d \in D$. Furthermore, the interpretation function I maps uncertainty intervals $[\alpha, \beta]$ in ULP expressions into properly defined DS models (i.e., into BBA or mass functions). ■

Definition 2 *(Uncertain FOL expressions).* Consider a quantifier-free first-order formula $\varphi(x)$ from a finite set of formulas Φ in some first-order language $\mathcal{L}$, with x being the only free variable in φ.[1] Then, an uncertain first-order logic expression relates the uncertainty associated with the truth of $\varphi(x)$ as:

$$\varphi(x), \text{ with uncertainty } [\alpha, \beta], \tag{3}$$

where $[\alpha, \beta]$ refers to the corresponding uncertainty interval, $0 \le \alpha \le \beta \le 1$, and x is interpreted over individuals in $D = \{d_1, \ldots, d_n\}$, with $n \ge 1$, i.e., $I(x) \in D$. The uncertain logic expression (3) can be abbreviated as $\varphi(x)_{[\alpha,\beta]}$. ■

The uncertainty interval $[\alpha, \beta]$ in Definition 2 indicates the support that we have for the expression $\varphi(x)$ being true or false. In addition, this interval can be used to characterize the level of ignorance that we have on the event $\varphi(x)$ being either true or false. In particular, the value of α quantifies the evidence or *belief* that we have on $\varphi(x)$ being true; β accounts for the *plausibility* of $\varphi(x)$ being true; and $\beta - \alpha$ accounts for the level of ignorance that we have on the event $\varphi(x)$ being either true or false. This concept also applies to groundings of ULP expressions, which are defined next.

[1] We assume a single free variable for ease of description. However, extending the definition to any number of finite variables in φ is straightforward. This extension will be used later in this paper, as new operators are introduced.

Definition 3 *(Grounded FOL expressions).* Consider an uncertain first-order logic expression $\varphi(x)_{[\alpha,\beta]}$ for $I(x) \in D = \{d_1, \ldots, d_n\}$, $n \geq 1$. The grounding of this expression for an individual $d_i \in D$, $i \in \{1, \ldots, n\}$, is represented by $(\varphi_I(I(x) = d_i))_{[\alpha,\beta]}$. When no confusion can arise, we may use the alternative notations $(\varphi_I(I(x) = d_i))_{[\alpha,\beta]} \equiv (\varphi(I(x) = d_i))_{[\alpha,\beta]} \equiv (\varphi(x/d_i))_{[\alpha,\beta]} \equiv (\varphi(d_i))_{[\alpha,\beta]}$. Note that the uncertainty of this grounding can be indicated explicitly (i.e., $\varphi(d_i)$, with uncertainty $[\alpha, \beta]$), or appended as a subindex of the grounded expression (e.g., $\varphi(d_i)_{[\alpha,\beta]}$). ■

To properly quantify the support that we have for each of the three possible events in $\varphi(d_i)$ (i.e., "true", "false", "do not know if either true/false"), we need to build a DS model for the expression $\varphi(d_i)_{[\alpha,\beta]}$. This model must be defined on a proper FoD. A proper FoD for the expression $\varphi(d_i)_{[\alpha,\beta]}$ is defined next. Its extension for the expression $\varphi(x)$ is straightforward.

Definition 4 *(Basic FoD).* Given a grounded logic expression $(\varphi(d))_{[\alpha,\beta]}$, with $d \in D$, a FoD is defined as:

$$\Theta_{\varphi(d)} = \{\varphi(d), \overline{\varphi(d)}\}, \tag{4}$$

where the first element (i.e., $\varphi(d)$) represents the event in which $\varphi(d)$ is true, and the second element (i.e., $\overline{\varphi(d)}$) represents the event in which $\varphi(d)$ is false. ■

An alternative definition of the basic FoD that may be appropriate for some applications appears in [20]. This alternative definition creates $\Theta_{\varphi(d)}$ as the cross product of $\Theta_{\varphi(d)}$ with a true–false set $\{\mathbf{1}, \mathbf{0}\}$, that is to say, $\Theta_{\varphi(d)} = \varphi(d) \times \{\mathbf{1}, \mathbf{0}\} = \{\varphi(d) \times \mathbf{1}, \varphi(d) \times \mathbf{0}\} = \{\varphi(d), \overline{\varphi(d)}\}$. Although not formally rigorous like the definitions in this manuscript, the alternative definition in [20] provides an intuitive way of generating FoDs and propagating uncertainties through DS-based logic inference.

When considering grounded logic expressions where more than one element of D becomes relevant, the basic FoD must be extended. An alternative is using Cartesian products of basic FoDs to obtain proper FoDs for this scenario. As an example, the comprehensive FoD, which is defined next, considers all the elements of D in a single FoD.

Definition 5 *(Comprehensive FoD).* Given a set of grounded logic expressions $\{(\varphi(d_1))_{[\alpha_1,\beta_1]}, (\varphi(d_2))_{[\alpha_2,\beta_2]}, \ldots, (\varphi(d_n)))_{[\alpha_n,\beta_n]}\}$, we define $\Theta_{\varphi(D)}$ to be the Cartesian product of all basic frames of discernment $\Theta_{\varphi(d_i)} = \{\varphi(d_i), \overline{\varphi(d_i)}\}$, $i = 1, 2, \ldots, n$, i.e., $\Theta_{\varphi(D)} = \{\{\varphi(d_1), \overline{\varphi(d_1)}\} \times \ldots \times \{\varphi(d_n), \overline{\varphi(d_n)}\}\}$. ■

Based on the FoDs in Definitions 4 and 5, we can model uncertain FOL expressions in the DS theoretic framework as follows.

Definition 6 *(DS theoretic model for uncertain logic expressions).* Consider an uncertain logic expression $\varphi(d)_{[\alpha,\beta]}$, with $d \in D$. Then, a DS theoretic model that would capture the uncertain information in this logic expression is the following mass assignment:

$$\begin{aligned} \varphi(d): \ & m(\varphi(d)) = \alpha; \\ & m(\overline{\varphi(d)}) = 1 - \beta; \\ & m(\Theta_{\varphi(d)}) = \beta - \alpha, \end{aligned} \tag{5}$$

defined over the basic FoD $\Theta_{\varphi(d)} = \{\varphi(d), \overline{\varphi(d)}\}$. When no confusion can arise, we may use the following notation, which shortens the parameter of the mass function by defining φ as a subindex of the mass function:

$$\begin{aligned} \varphi(d): \ & m_{\varphi}(d) = \alpha; \\ & m_{\varphi}(\overline{d}) = 1 - \beta; \\ & m_{\varphi}(\Theta) = \beta - \alpha. \quad ■ \end{aligned} \tag{6}$$

Note that, since the DS model for uncertain logic expressions is defined over a dichotomous FoD (i.e., the FoD has only two elements), it can be completely characterized by an uncertainty interval. Then, in the following, we may use $m(\varphi(d)) = [\alpha, \beta]$ as a shorter notation for the DS model in Definition 6.

4. Definitions of negation, conjunction, and disjunction in ULP

In this section we introduce the basic operators for ULP, which are defined using generic DS fusion operations. The definition of operators in this section is done using generic DS fusion operators. Then, in Section 5, we show how to select specific fusion operators to obtain a desired behavior. For example, one may be interested in attaining consistency with classical logic (the case thoroughly discussed in Section 5), or one may want to relax this condition and model a paraconsistent logic (a case left for future work on extensions of ULP).

4.1. Uncertain logic negation

The negation operation in ULP is based on the definition of set complement as it applies to DS models (see [30], for example). Using this approach, we define the complement of a BBA, and then define the logical not of this BBA based on said complement. This is described next.

Definition 7 *(Complementary BBA).* Consider a basic FoD $\Theta_{\varphi(d)} = \{\varphi(d), \overline{\varphi(d)}\}$, and a BBA $m_\varphi(\cdot)$ defined as:

$$m_\varphi(d) = \alpha;\;\; m_\varphi(\overline{d}) = 1-\beta;\;\; m_\varphi(\Theta) = \beta - \alpha. \tag{7}$$

A complementary BBA for (7) is given by [30]:

$$m^c_\varphi(d) = 1-\beta;\; m^c_\varphi(\overline{d}) = \alpha;\; m^c_\varphi(\Theta) = \beta - \alpha. \quad \blacksquare \tag{8}$$

Based on the complementary BBA, we can define an uncertain logic negation as follows.

Definition 8 *(Logical not in ULP).* Consider an uncertain logic expression $\varphi(d)_{[\alpha,\beta]}$. Also, consider its corresponding DS model, which is defined by (6). Then, the ULP-negation of $\varphi(d)_{[\alpha,\beta]}$ denoted $\neg(\varphi(d)_{[\alpha,\beta]})$ is defined as $(\neg\varphi(d))_{[1-\beta,1-\alpha]}$. We utilize the complementary BBA corresponding to (6) as the DS theoretic model for $\neg\varphi(d)$, i.e.,

$$\begin{aligned}\neg\varphi(d):\;\; m^c_\varphi(d) &= 1-\beta;\\ m^c_\varphi(\overline{d}) &= \alpha;\\ m^c_\varphi(\Theta) &= \beta-\alpha. \quad \blacksquare\end{aligned} \tag{9}$$

It is clear that the complementary BBA associated with $\varphi_{[\alpha,\beta]}(x)$ is $\neg\varphi_{[1-\beta,1-\alpha]}(x)$.

4.2. Logical AND/OR in ULP

Definition 9 *(Logical AND & OR in ULP).* Suppose that we have M propositions $(\varphi_i(d))_{[\alpha_i,\beta_i]}, i = \{1,2,\ldots,M\}$. Their corresponding ULP models are $\varphi_i(d):\;\; m_{\varphi_i}(d) = \alpha_i;\;\; m_{\varphi_i}(\overline{d}) = 1-\beta_i;\;\; m_{\varphi_i}(\Theta_{\varphi_i(d)}) = \beta_i - \alpha_i$, for $i = 1,2,\ldots,M$. We propose to utilize the following DS theoretic models for the logical AND and OR of the statements $(\varphi_i(d))_{[\alpha_i,\beta_i]}, i = \{1,2,\ldots,M\}$:

$$\begin{aligned}&\bigwedge_{i=1}^{M}\varphi_i(d):\;\; m(\cdot) = \bigcap_{i=1}^{M} m_{\varphi_i}(\cdot);\\ \text{and}\quad &\bigvee_{i=1}^{M}\varphi_i(d):\;\; m(\cdot) = \left(\bigcap_{i=1}^{M} m^c_{\varphi_i}(\cdot)\right)^c,\end{aligned} \tag{10}$$

where $\bigcap$ denotes an appropriate fusion operator. $\blacksquare$

Remarks:

1) The definition of the conjunction in Definition 9 explicitly includes its corresponding BBA. This BBA is obtained from the direct application of the ULP extension of the definition of conjunction, that is to say, from defining $\varphi_1(d) \vee \varphi_2(d) \vee \ldots \vee \varphi_M(d) \equiv \neg(\neg\varphi_1(d) \wedge \neg\varphi_2(d) \wedge \ldots \wedge \neg\varphi_M(d))$.
2) Based on Definition 9, we can define the AND/OR operators for unquantified expressions $\varphi_i(x)$, with uncertainty $[\alpha_i, \beta_i]$, $i = 1,\ldots,M$, as: $\bigwedge_{i=1}^{M}\varphi_i(x):\;\; m(\cdot) = \bigcap_{i=1}^{M} m_{\varphi_i}(\cdot)$; and $\bigvee_{i=1}^{M}\varphi_i(x): m(\cdot) = \left(\bigcap_{i=1}^{M} m^c_{\varphi_i}(\cdot)\right)^c$, for the AND and OR operations, respectively.
3) A model similar to the one in Definition 9 can be obtained for the case of AND/OR operations of a set of expressions $\{\varphi(x_i)\}$ with uncertainty $[\alpha_i, \beta_i]$,
$x_i \in \{x_1, x_2, \ldots, x_k\}$. In this case: $\bigwedge_{i=1}^{k}\varphi(x_i): m(\cdot) = \bigcap_{i=1}^{k} m_\varphi(\cdot)$, and $\bigvee_{i=1}^{k}\varphi(x_i): m(\cdot) = \left(\bigcap_{i=1}^{k} m^c_\varphi(\cdot)\right)^c$. This case represents AND/OR models applied to the truthfulness of elements $\{x_i\}$ satisfying a property φ, whereas (10) analyzes the case of x satisfying multiple properties $\{\varphi_i\}$.
4) Definition 9 also extends to AND/OR operations of set of expressions $\{\varphi(d_i)\}$ with uncertainty $[\alpha_i, \beta_i]$,
$d_i \in \{d_1, d_2, \ldots, d_n\}$. In this case: $\bigwedge_{i=1}^{n}\varphi(d_i): m(\cdot) = \bigcap_{i=1}^{n} m_\varphi(\cdot)$, and $\bigvee_{i=1}^{n}\varphi(d_i): m(\cdot) = \left(\bigcap_{i=1}^{n} m^c_\varphi(\cdot)\right)^c$.

4.3. Other ULP operators

It is possible to extend the ULP operators described above and create new operators. As an example, consider implication rules. An uncertain logic implication can be defined by extending the (classical) definition for the implication rule based on AND/OR operators to the uncertain logic framework. The classical logic definition is: Given two statements $\varphi_1(\cdot)$ and $\varphi_2(\cdot)$, an implication rule has the property:

$$\varphi_1(d_i) \implies \varphi_2(d_j) \equiv \neg\varphi_1(d_i) \vee \varphi_2(d_j) \\ \equiv \neg\left(\varphi_1(d_i) \wedge \neg\varphi_2(d_j)\right),$$

where $d_i, d_j \in D$. Now we define an uncertain implication rule as follows.

Definition 10 *(Uncertain implication rule).* Consider an antecedent $\varphi_1(d_i)$ and a consequent $\varphi_2(d_j)$, with uncertainty intervals $[\alpha_{\varphi_1(d_i)}, \beta_{\varphi_1(d_i)}]$ and $[\alpha_{\varphi_2(d_j)}, \beta_{\varphi_2(d_j)}]$, respectively. Furthermore, suppose that said uncertainty is represented via the DS theoretic models $m_1(\cdot)$ and $m_2(\cdot)$ over the FoDs $\Theta_{\varphi_1(d_i)}$ and $\Theta_{\varphi_2(d_j)}$, respectively. Then, the implication rule $\varphi_1(\cdot) \implies \varphi_2(\cdot)$ is taken to have the following DS model:

$$m_{\varphi_1\to\varphi_2}(\cdot) = (m_1^c \vee m_2)(\cdot) = (m_1 \wedge m_2^c)^c(\cdot), \tag{11}$$

over the FoD $\Theta_{\varphi_1(d_i)} \times \Theta_{\varphi_2(d_j)}$. ■

5. Attaining consistency with classical logic

Recall that ULP operators can be tuned to satisfy particular properties. This can be accomplished by selecting an appropriate fusion operator $\bigcap$ in (10). Note that this operator directly impacts the behavior of logic operations, as many logic operations are derived from the fundamental AND and OR operators. This fusion operator must be then selected carefully to obtain the particular properties that we wish to obtain in an uncertainty measuring and tracking system.

One fundamental configuration that we may wish to attain for ULP is one that is consistent with classical logic. In this section we analyze the selection of the appropriate fusion operator for this case. To ensure consistency with classical logic, we are interested in a fusion operator that allows the logic operators to satisfy the following properties:

(a) **Double negation.** Given a proposition $\varphi(d)$, the BBA corresponding to its double-negation is the same model as the one associated with $\varphi(d)$. By definition of the negation in ULP (see Definition 8), ULP already satisfies $\neg\neg\varphi(d) = \varphi(d)$).
(b) **De Morgan's laws.** $(\varphi_1(d) \vee \varphi_2(d))$ and $\neg(\neg\varphi_1(d) \wedge \neg\varphi_2(d))$ have identical DS theoretic models. Also, $(\varphi_1(d) \wedge \varphi_2(d))$ and $\neg(\neg\varphi_1(d) \vee \neg\varphi_2(d))$ have identical DS theoretic models. By definition of the conjunction and disjunction in ULP (see Definition 9), ULP already satisfies the De Morgan's laws.
(c) **Idempotency.** Uncertain logic AND and OR operators are idempotent, i.e.: $\varphi(d)_{[\alpha,\beta]} \wedge \varphi(d)_{[\alpha,\beta]} = \varphi(d)_{[\alpha,\beta]} \vee \varphi(d)_{[\alpha,\beta]} = \varphi(d)_{[\alpha,\beta]}$.
(d) **Commutativity.** Uncertain logic AND and OR operators are commutative, i.e.: $\varphi_1(d) \wedge \varphi_2(d) = \varphi_2(d) \wedge \varphi_1(d)$, and $\varphi_1(d) \vee \varphi_2(d) = \varphi_2(d) \vee \varphi_1(d)$.
(e) **Associativity.** Uncertain logic AND and OR operators must be associative. i.e.: $\varphi_1(d) \wedge [\varphi_2(d) \wedge \varphi_3(d)] = [\varphi_1(d) \wedge \varphi_2(d)] \wedge \varphi_3(d)$, and $\varphi_1(d) \vee [\varphi_2(d) \vee \varphi_3(d)] = [\varphi_1(d) \vee \varphi_2(d)] \vee \varphi_3(d)$.
(f) **Distributivity.** Uncertain logic AND and OR operators must be distributive, i.e.: $\varphi_1(d) \wedge [\varphi_2(d) \vee \varphi_3(d)] = [\varphi_1(d) \wedge \varphi_2(d)] \vee [\varphi_1(d) \wedge \varphi_3(d)]$, and $\varphi_1(d) \vee [\varphi_2(d) \wedge \varphi_3(d)] = [\varphi_1(d) \vee \varphi_2(d)] \wedge [\varphi_1(d) \vee \varphi_3(d)]$.
(g) **Consistent Boolean models.** In the absence of uncertainty, ULP models converge to those of Boolean logic. That is to say, if $\alpha, \beta \in \{0, 1\}$, and $\alpha = \beta$, inference results in ULP render intervals that are either $[0, 0]$ or $[1, 1]$, and that correspond to the false/true truth value assignments that would be obtained using Boolean logic.
(h) **Consistent probabilistic models.** In probabilistic scenarios (i.e., where $\alpha = \beta$ for every uncertainty interval $[\alpha, \beta]$), ULP inference renders probabilistic models.
(i) **Uniqueness of model.** Given propositions $\varphi_1(d)_{[\alpha_1,\beta_1]}$ and $\varphi_2(d)_{[\alpha_2,\beta_2]}$, with $\alpha_1 \neq \alpha_2$ and $\beta_1 \neq \beta_2$ the AND and OR operations satisfy $\varphi_1(d) \wedge \varphi_2(d) \neq \varphi_1(d) \vee \varphi_2(d)$.

Next we show that ULP does not satisfy all these properties when we use DRC. Then we present the CFE as a better alternative for classically consistent ULP.

5.1. DCR-based ULP

Consider the two-propositions (i.e., $M = 2$) case for ULP AND/OR operations. Table 1 contains the DCR-based logical AND and OR operations for this case. Notice that the mass assignments for the AND operation (i.e., $\varphi_1(d) \wedge \varphi_2(d)$) are exactly the same as the ones obtained for the OR operation (i.e., $\varphi_1(d) \vee \varphi_2(d)$). Having identical models for both AND and OR operators suggests that, although DCR may work as a fusion operator for certain operations, it does not render models that

Table 1
DCR-based logical AND and OR. In both cases, the masses should be normalized by $1 - K$, with $K = 1 - \sum_{A \in \mathcal{F}} m(A) = \alpha_1(1-\beta_2) + (1-\beta_1)\alpha_2$. Note that the DS models for AND and OR are identical, which suggests that DCR is not an appropriate fusion operator for classically consistent ULP operations.

Focal set	$\varphi_1(d) \wedge \varphi_2(d)$	$\varphi_1(d) \vee \varphi_2(d)$
d	$\alpha_1\beta_2 + (\beta_1 - \alpha_1)\alpha_2$	$\alpha_1\beta_2 + (\beta_1 - \alpha_1)\alpha_2$
$\overline{d}$	$(1-\beta_1)(1-\alpha_2) + (\beta_1 - \alpha_1)(1-\beta_2)$	$(1-\beta_1)(1-\alpha_2) + (\beta_1 - \alpha_1)(1-\beta_2)$
$\Theta_{\varphi_1,\varphi_2,d}$	$(\beta_1 - \alpha_1)(\beta_2 - \alpha_2)$	$(\beta_1 - \alpha_1)(\beta_2 - \alpha_2)$

satisfy important properties for all the logical operations defined above. Namely, DCR-based uncertain logic does not satisfy property (i) above. As an alternative, we propose using the CFE, which is analyzed next.

5.2. CFE-based uncertain logic

Recall (from Section 2) that CFE-based fusion requires the definition of a set of coefficients γ_i. We define the Logic Consistent (LC) strategy for the definition of the CFE coefficients as follows.

Definition 11 *(Logic Consistent (LC) strategy).* Consider the case $M = 2$ in (10), and let us define $\underline{\alpha} = \min(\alpha_1, \alpha_2)$; $\underline{\beta} = \min(\beta_1, \beta_2)$; $\overline{\alpha} = \max(\alpha_1, \alpha_2)$; $\overline{\beta} = \max(\beta_1, \beta_2)$; $\delta_1 = \beta_1 - \alpha_1$; $\delta_2 = \beta_2 - \alpha_2$; $\underline{\delta} = \underline{\beta} - \underline{\alpha}$; and $\overline{\delta} = \overline{\beta} - \overline{\alpha}$. Then select the CFE coefficients as follows:

$$\gamma_1(d) = \gamma_2(d) \equiv \gamma(d); \quad \gamma_1(\overline{d}) = \gamma_2(\overline{d}) \equiv \gamma(\overline{d});$$
$$\text{and} \quad \gamma_1(\Theta) = \gamma_2(\Theta) \equiv \gamma(\Theta),$$

where $\gamma(d)$, $\gamma(\overline{d})$, and $\gamma(\Theta)$ are given by:

a. Logical AND:

If $\delta_1 + \delta_2 \neq 0$:
$$\gamma(d) = \frac{\underline{\alpha}(\beta_1+\beta_2) - \underline{\beta}(\alpha_1+\alpha_2)}{2(\delta_1+\delta_2)};$$
$$\gamma(\overline{d}) = \frac{1}{2} - \frac{\underline{\beta}(2-\alpha_1-\alpha_2) - \underline{\alpha}(2-\beta_1-\beta_2)}{2(\delta_1+\delta_2)};$$
$$\gamma(\Theta) = \frac{\underline{\delta}}{\delta_1+\delta_2}.$$
If $\delta_1 + \delta_2 = 0$:
$$\gamma(d) = \frac{\underline{\alpha} - \gamma(\Theta)\,(\alpha_1+\alpha_2)}{2};$$
$$\gamma(\overline{d}) = \frac{(1-\underline{\alpha}) - \gamma(\Theta)\,(2-\alpha_1-\alpha_2)}{2};$$
$$\gamma(\Theta) \quad \text{is arbitrary in the interval } [0,1].$$

b. Logical OR:

If $\delta_1 + \delta_2 \neq 0$:
$$\gamma(d) = \frac{1}{2} - \frac{\overline{\beta}(2-\alpha_1-\alpha_2) - \overline{\alpha}(2-\beta_1-\beta_2)}{2(\delta_1+\delta_2)};$$
$$\gamma(\overline{d}) = \frac{\overline{\alpha}(\beta_1+\beta_2) - \overline{\beta}(\alpha_1+\alpha_2)}{2(\delta_1+\delta_2)};$$
$$\gamma(\Theta) = \frac{\overline{\delta}}{\delta_1+\delta_2}.$$
If $\delta_1 + \delta_2 = 0$:
$$\gamma(d) = \frac{\overline{\alpha} - \gamma(\Theta)\,(\alpha_1+\alpha_2)}{2};$$

$$\gamma(\bar{d}) = \frac{(1-\overline{\alpha}) - \gamma(\Theta)\,(2-\alpha_1-\alpha_2)}{2};$$
$$\gamma(\Theta) \quad \text{is arbitrary in the interval } [0,1]. \quad \blacksquare$$

Based on the LC Strategy, we can define the CFE-based AND/OR operators, as well as implication rules, as follows.

5.2.1. CFE-based AND and OR operators

Given two uncertain propositions $\varphi_1(d)_{[\alpha_1,\beta_1]}$ and $\varphi_2(d)_{[\alpha_2,\beta_2]}$, the LC strategy renders the following BBA for the AND operation (see proof in Appendix A):

$$\begin{aligned}\varphi_1(d)\wedge\varphi_2(d):\quad & m(d) &&= \underline{\alpha};\\ & m(\bar{d}) &&= 1-\underline{\beta};\ \text{and}\\ & m(\Theta_{(\varphi_1\wedge\varphi_2)(d)}) &&= \underline{\beta}-\underline{\alpha},\end{aligned} \tag{12}$$

with $\underline{\alpha} = \min(\alpha_1,\alpha_2)$ and $\underline{\beta} = \min(\beta_1,\beta_2)$. Similarly, when used for the OR operation, the LC strategy renders the following BBA:

$$\begin{aligned}\varphi_1(d)\vee\varphi_2(d):\quad & m(d) &&= \overline{\alpha};\\ & m(\bar{d}) &&= 1-\overline{\beta};\ \text{and}\\ & m(\Theta_{(\varphi_1\vee\varphi_2)(d)}) &&= \overline{\beta}-\overline{\alpha},\end{aligned} \tag{13}$$

with $\overline{\alpha} = \max(\alpha_1,\alpha_2)$ and $\overline{\beta} = \max(\beta_1,\beta_2)$.

It can be proven that CFE-based fusion using the LC strategy satisfies the properties (a)–(g) above. As mentioned above, properties (a) and (b) are satisfied by the basic definitions of ULP models. A proof for properties (c)–(f) can be found in [20], and is presented in Appendix B. Property (h) can be easily proven as follows: Consider uncertainty parameters defined as $\alpha_1 = \beta_1$ and $\alpha_2 = \beta_2$. Let us denote $\varphi_1(d)_{[\alpha_1,\beta_1]}$ and $\varphi_2(d)_{[\alpha_2,\beta_2]}$ as $\varphi(d)_{[\alpha_1]}$ and $\varphi(d)_{[\alpha_2]}$, respectively. We then get

$$\begin{aligned}&\varphi(d)_{[\alpha_1]}\wedge\varphi(d)_{[\alpha_2]} = \varphi(d)_{[\underline{\alpha}]};\\ &\varphi(d)_{[\alpha_1]}\vee\varphi(d)_{[\alpha_2]} = \varphi(d)_{[\overline{\alpha}]},\end{aligned} \tag{14}$$

where $\underline{\alpha} = \min(\alpha_1,\alpha_2)$ and $\overline{\alpha} = \max(\alpha_1,\alpha_2)$. Property (g) can be easily proved by building a truth table based on the models for (h). Finally, property (i) is satisfied by the models (12) and (13) above. ULP models for multiple propositions can be found in Appendix C.

5.2.2. CFE-based implication

In the case of logic implications, the model in Definition 10 renders the following BBA:

$$\begin{aligned}&\varphi_1(d_i) \implies \varphi_2(d_j):\\ &\quad m_{\varphi_1\to\varphi_2}(d_i\times\Theta_{\varphi_2,d_j}) &&= \tfrac{1}{2}\alpha_R;\\ &\quad m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times d_j) &&= \tfrac{1}{2}\alpha_R;\\ &\quad m_{\varphi_1\to\varphi_2}(\overline{d_i}\times\Theta_{\varphi_2,d_j}) &&= \tfrac{1}{2}(1-\beta_R);\\ &\quad m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times\overline{d_j}) &&= \tfrac{1}{2}(1-\beta_R);\\ &\quad m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times\Theta_{\varphi_2,d_j}) &&= \beta_R-\alpha_R,\end{aligned} \tag{15}$$

with $\alpha_R = \max(1-\beta_1,\alpha_2)$ and $\beta_R = \max(1-\alpha_1,\beta_2)$. Note that α_R and β_R define the uncertainty interval $[\alpha_R,\beta_R]$ of the implication rule. This interval is obtained from projecting the BBA defined in (15) into the true–false components of the original BoEs, which we label as $\{\mathbf{1},\mathbf{0}\}$, for ease of notation, as follows:

$$\begin{aligned}m_{\varphi_1\to\varphi_2}(\mathbf{1}) &= m_{\varphi_1\to\varphi_2}(d_i\times\Theta_{\varphi_2,d_j})\\ &\quad + m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times d_j) = \alpha_R;\\ m_{\varphi_1\to\varphi_2}(\mathbf{0}) &= m_{\varphi_1\to\varphi_2}(\overline{d_i}\times\Theta_{\varphi_2,d_j})\\ &\quad + m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times\overline{d_j}) = 1-\beta_R;\\ m_{\varphi_1\to\varphi_2}(\{\mathbf{1},\mathbf{0}\}) &= m_{\varphi_1\to\varphi_2}(\Theta_{\varphi_1,d_i}\times\Theta_{\varphi_2,d_j}) = \beta_R-\alpha_R.\end{aligned} \tag{16}$$

The **1** component corresponds to the event "$\varphi_1 \to \varphi_2$", and the **0** component corresponds to the event "$\overline{\varphi_1 \to \varphi_2}$". Note that, in the Boolean case (i.e., $\alpha_1 = \beta_1 \in \{\mathbf{1},\mathbf{0}\}$ and $\alpha_2 = \beta_2 \in \{\mathbf{1},\mathbf{0}\}$), the CFE-based uncertain implication rule converges to the conventional logic result (see Table 2).

Table 2
CFE-based implication with Boolean arguments. Uncertainty parameters are defined so that they represent complete certainty on the truth (or falsity) of each proposition. In this case, there is complete certainty of the truth of the output model (as occurs with classical logic.)

Parameters		Uncertainty of the rule
$[\alpha_1, \beta_1]$	$[\alpha_2, \beta_2]$	$[\alpha_R, \beta_R]$
[0, 0]	[0, 0]	[1, 1]
[0, 0]	[1, 1]	[1, 1]
[1, 1]	[0, 0]	[0, 0]
[1, 1]	[1, 1]	[1, 1]

5.3. ULP as a model for capturing variable granularities of many-valued logics

ULP naturally adapts to different quantizations in the uncertainty of the evidence. For example, if all the uncertainty intervals $[\alpha, \beta]$ in the premises are such that $\alpha, \beta \in \{0, 1\}$, then the uncertainty results provided by ULP will be also in $\{0, 1\}$. In general, if uncertainties of the premises are quantized in steps of $1/n$, with $n = 2, 3, \ldots$, then the result will also be quantized by this step size.

To illustrate this property, consider an n-ary logic in which we have the propositions φ_i and φ_j, with $i, j \in \{0, 1, \ldots, n\}$ and $n \geq 1$, such that the uncertainty of the premises is probabilistic and modeled as follows:

$$\varphi_i, \text{ with uncertainty } \frac{i}{n}; \text{ and } \varphi_j, \text{ with uncertainty } \frac{j}{n}. \tag{17}$$

Now consider an uncertain implication rule $\varphi_i \implies \varphi_j$. Given (17), the uncertainty of this implication is given by:

$$\varphi_i \implies \varphi_j, \text{ with uncertainty } \max(1 - \frac{i}{n}, \frac{j}{n}). \tag{18}$$

Note that the resulting uncertainty in the implication rule is a multiple of $1/n$. It can be shown that the same behavior occurs for NOT, AND, and OR operations. Hence, the ULP framework can be used to model an n-ary (i.e., many-valued) logic, where we can have from coarse ($n = 2$, Boolean logic) to infinitely small quantizations (when $n \to \infty$). Moreover, we can increase the granularity dynamically during an inference process, right at the moment when new evidence introduces a refined granularity. That is to say, ULP can be used to adaptively adjust the quantization of the uncertain models according to the input data.

6. Quantifiers in ULP

6.1. Existential quantifiers

Existential quantifiers are used to express that a sentence is true for at least one object of the universe of discourse. We define existential quantifiers in ULP as follows.

Definition 12 *(Existential quantifier in ULP).* Consider the statement:

$$(\exists x \;\; \varphi(x))_{[\alpha,\beta]}, \tag{19}$$

where $[\alpha, \beta]$ refers to the corresponding uncertainty interval, $0 \leq \alpha \leq \beta \leq 1$, and x is interpreted over elements in $D = \{d_1, d_2, \ldots, d_n\}$, with $n \geq 1$. Let us define an FoD over the domain D as $\Theta_{\varphi,D} = \Theta_{\varphi,d_1} \times \Theta_{\varphi,d_2} \times \ldots \times \Theta_{\varphi,d_n}$. Then, we define the DS theoretic model for (19) as:

$$\bigvee_{i=1}^{n} (\varphi(d_i))_{[\alpha_i,\beta_i]}, \tag{20}$$

over the FoD $\Theta_{\varphi,D}$, subject to the constraint:

$$\begin{aligned} m(\mathbf{1}) &= \textstyle\sum_{i=1}^{n} m_\varphi(d_i) = \alpha; \\ m(\mathbf{0}) &= \textstyle\sum_{i=1}^{n} m_\varphi(\overline{d_i}) = 1 - \beta; \\ m(\Theta_{\varphi,\Theta_D}) &= \beta - \alpha, \end{aligned} \tag{21}$$

where we have adopted the notation d_i and $\overline{d_i}$ to represent the sets $\{\ldots \times \Theta_{\varphi,d_{i-2}} \times \Theta_{\varphi,d_{i-1}} \times \varphi(d_i) \times \Theta_{\varphi,d_{i+1}} \times \Theta_{\varphi,d_{i+2}} \times \ldots\}$, and $\{\ldots \times \Theta_{\varphi,d_{i-2}} \times \Theta_{\varphi,d_{i-1}} \times \overline{\varphi(d_i)} \times \Theta_{\varphi,d_{i+1}} \times \Theta_{\varphi,d_{i+2}} \times \ldots\}$, respectively. ■

Remarks:

1) Definition 12 is based on the substitutional conception of quantifiers, according to which ∃ is treated as a generalization of the disjunction [7] grounded over the finite set $D = \{d_1, d_2, \ldots, d_n\}$.
2) When using the CFE-based ULP model described above, the constraint (21) is satisfied if the uncertainty of at least one of the propositions $\varphi(d_i)$ in (20) is $[\alpha, \beta]$, and the uncertainty of every other proposition is $[0, 0]$ (or, in general, $[\alpha_j, \beta_j]$, with $\alpha_j \leq \alpha$, $\beta_j \leq \beta$, and $i \neq j$), then the DS model corresponding to (20) is equivalent to the DS model corresponding to (19) when the OR operations are computed as indicated by Definitions 9 and 11.

6.2. Universal quantifiers

Universal quantifiers are used to express that a sentence is true for every object of the universe of discourse. We define universal quantifiers in ULP as follows.

Definition 13 *(Universal quantifier in ULP).* Consider the statement:

$$(\forall x \;\; \varphi(x))_{[\alpha,\beta]}, \tag{22}$$

where $[\alpha, \beta]$ refers to the corresponding uncertainty interval, $0 \leq \alpha \leq \beta \leq 1$, and x is interpreted over elements in $D = \{d_1, d_2, \ldots, d_n\}$, with $n \geq 1$. Let us define an FoD over the domain D as $\Theta_{\varphi,D} = \Theta_{\varphi,d_1} \times \Theta_{\varphi,d_2} \times \ldots \times \Theta_{\varphi,d_n}$. Then, we define the DS theoretic model for (22) as:

$$\bigwedge_{i=1}^{n} (\varphi(d_i))_{[\alpha_i,\beta_i]}, \tag{23}$$

over the FoD $\Theta_{\varphi,D}$, subject to the constraint:

$$\begin{aligned} m(\mathbf{1}) &= \sum\nolimits_{i=1}^{n} m_\varphi(d_i) = \alpha; \\ m(\mathbf{0}) &= \sum\nolimits_{i=1}^{n} m_\varphi(\overline{d_i}) = 1 - \beta; \\ m(\Theta_{\Theta_{\varphi,\Theta_D}}) &= \beta - \alpha, \end{aligned} \tag{24}$$

where we have adopted the notation d_i and $\overline{d_i}$ to represent the sets $\{\ldots \times \Theta_{\varphi,d_{i-2}} \times \Theta_{\varphi,d_{i-1}} \times \varphi(d_i) \times \Theta_{\varphi,d_{i+1}} \times \Theta_{\varphi,d_{i+2}} \times \ldots\}$, and $\{\ldots \times \Theta_{\varphi,d_{i-2}} \times \Theta_{\varphi,d_{i-1}} \times \overline{\varphi(d_i)} \times \Theta_{\varphi,d_{i+1}} \times \Theta_{\varphi,d_{i+2}} \times \ldots\}$, respectively. ■

Remarks:

1) Definition 13 is based on the substitutional conception of quantifiers, according to which ∀ is treated as a generalization of the conjunction [7] grounded over the finite set $D = \{d_1, d_2, \ldots, d_n\}$. This definition is justified as long as the conjunction satisfies both associativity and commutativity, as is the case of the CFE-based ULP disjunction described above.
2) When using the CFE-based ULP model described above, the constraint (24) is satisfied if the uncertainty of every proposition $\varphi(d_i)$ in (23) is $[\alpha, \beta]$.
3) Although an infinite number of solutions satisfy (24), a useful solution (e.g., for universal instantiation on inference) is given by $m_\varphi(d_i) = \alpha$; $m_\varphi(\overline{d_i}) = 1 - \beta$; and $m_\varphi(\{d_i, \overline{d_i}\}) = \beta - \alpha, i = 1, 2, \ldots, n$. This solution can be proven by applying idempotency to the AND operator.

7. Inference in ULP

When using the LC strategy for CFE-based ULP, inference in ULP shares the fundamental principles of classical logic. Inheriting inference rules and algorithms from classical logic is possible due to the definition of ULP as an algebra that reproduces the core properties of classical logic.

7.1. Modus Ponens (MP)

Modus Ponens (MP) rule states that, whenever the sentences $\varphi \implies \psi$ and φ have been established, then we can infer the sentence ψ as well. MP extends to ULP as follows. Consider:

$$\begin{aligned} &\varphi_1(d_1), \text{with uncertainty } [\alpha_1, \beta_1]; \\ &\varphi_2(d_2), \text{with uncertainty } [\alpha_2, \beta_2]; \text{ and} \\ \varphi_1(d_1) \implies &\varphi_2(d_2), \text{with uncertainty } [\alpha_R, \beta_R]. \end{aligned} \tag{25}$$

Then, given the uncertain premises $\varphi_1(d_1) \implies \varphi_2(d_2)$ and $\varphi_1(d_1)$, MP allows us to infer the uncertain expression $\varphi_2(d_2)$. Note that if the uncertainty parameters $[\alpha_2, \beta_2]$ are unknown, their value should be obtained by using the definition of uncertain implication rules in Section 5.2.2 above. Based on (15), if we know a model $m_{\varphi_1 \to \varphi_2}$ for the implication rule, as well as a model m_{φ_1} for the antecedent, we could obtain a model for an unknown consequent $m_2(\cdot)$. In this case:

$$\alpha_2 = \begin{cases} \alpha_R, & \text{if } \alpha_R > 1 - \beta_1; \\ [0, \alpha_R], & \text{if } \alpha_R = 1 - \beta_1; \text{ and} \\ \text{no solution}, & \text{otherwise}, \end{cases} \tag{26}$$

and

$$\beta_2 = \begin{cases} \beta_R, & \text{if } \beta_R > 1 - \alpha_1; \\ [0, \beta_R], & \text{if } \beta_R = 1 - \alpha_1; \text{ and} \\ \text{no solution}, & \text{otherwise}. \end{cases} \tag{27}$$

Some important observations:

- Given a pair of uncertainty intervals $[\alpha_1, \beta_1]$ and $[\alpha_R, \beta_R]$, it is not always possible to infer anything about a model for the consequent (i.e., $m_2(\cdot)$). This is consistent with the application of the Modus Ponens (MP) rule, according to which, if we know $\varphi_1 \implies \varphi_2$, and also that φ_1 is true, then we can infer that φ_2 is true. However, if we know that φ_1 is false (i.e., $\neg\varphi_1$ is true), we cannot say anything about the truth value of φ_2.
- Furthermore, once we have enough support in $m_1(\cdot)$ to infer something regarding $m_2(\cdot)$, the only conclusion that we can provide regarding the uncertainty of $m_2(\cdot)$ is that said uncertainty is $m_R(\cdot)$. That is, after we have gathered a certain amount of evidence regarding the truth of the antecedent, getting more evidence is not going to affect the confidence we have on the truth of the consequent (i.e., α_2 is bounded by α_R, and β_2 is bounded by β_R).
- When $\alpha_R = 1 - \beta_1$ and $\beta_R = 1 - \alpha_1$, an infinite number of solutions exist for $[\alpha_2, \beta_2]$. We could use the minimum commitment criterion to decide $\alpha_2 = 0$ and $\beta_2 = \beta_R$.

To better understand MP in ULP, consider an example where $\alpha_1 = \beta_1 = \alpha_2 = \beta_2 = 1$. In this case, we obtain $\alpha_R = \beta_R = 1$. Furthermore, given the $\varphi_1 \implies \varphi_2$ and φ_1, then we can infer φ_2 with uncertainty $[\alpha_2 = \beta_2] = [1, 1]$. This case represents a scenario with no uncertainty.

Now consider a scenario where there is uncertainty in the rule, in such a way that $[\alpha_R, \beta_R] = [0.5, 1.0]$, and assume that we have a model for the uncertainty of φ_1 such that $\alpha_1 = \beta_1 = 1$. Then, MP allows us to infer φ_2, with the uncertainty $[\alpha_2, \beta_2]$ obtained from the equations $\alpha_R = \max(1 - \beta_1, \alpha_2)$ and $\beta_R = \max(1 - \alpha_1, \beta_2)$. Solving these equations we obtain $\alpha_2 = 0.5$ and $\beta_2 = 1$.

7.2. Other rules of inference

As with MP, ULP can be extended by incorporating new rules of inference that already exist in conventional logic inference. Some examples of new rules of inference are: *Modus Tolens* (MT), AND elimination (AE), AND introduction (AI), universal instantiation (UI), and existential instantiation (EI). The definition of these rules of inference is straightforward based on their definition for conventional logic, and is not included in this manuscript. For a description of these rules of inference, we refer the reader to [28].

7.3. Example: applying inference rules in ULP

Consider the following problem, extracted from [28]. The law says that it is a crime to sell an unregistered gun. Red has several unregistered guns, and all of them were purchased from Lefty. Based on these premises, can we derive the conclusion that Lefty is a criminal? Moreover, how is this conclusion affected if these premises become uncertain?

Table 3 contains the FOL representation of these premises, as well as a derivation of the ULP-quantified conclusion “Lefty is a criminal”. Note that, if all the rules and premises are true (i.e., Boolean scenario with true clauses, which are represented by an uncertainty interval $[1, 1]$), the conclusion effectively shows that Lefty is a criminal. However, an uncertainty interval starts growing in this conclusion when the premises lose certainty. The fourth column in Table 3 shows how uncertainty is tracked in this problem. Note how in the Boolean scenario the conclusion matches that of classical logic. Also, note that if the inputs (clauses 1–3) are probabilistic, the result is probabilistic too.

Fig. 1 illustrates a more general ULP scenario, where the lower bound of the conclusion’s uncertainty (i.e., α_{10}) is a function of the input arguments α_1 and β_2. By looking at this figure, we can identify a region (defined by $\alpha_1 + \beta_2 \leq 1$) for which we cannot derive the uncertainty of the conclusion. This is expected, given that, as mentioned above, we cannot always infer something from a MP rule when we have not enough evidence for the antecedent. In addition, in the figure we can see how α_{10} increases as the evidence of the input premises increases (i.e., α_1 and β_2 increase).

Table 3
Premises and inference process for the derivation of the conclusion "Is Lefty a criminal?". The example's premises and inference process are introduced in [28] (note that "WeaponX" is a Skolem constant). Uncertainty modeling and tracking is done using the methods introduced in this paper. Inference rule abbreviations are: △ (premise), EI (Existential Instantiation), UI (Universal Instantiation), MP (Modus Ponens), and AU (And Elimination).

	Logic expression	Inference rule	Uncertainty interval
1	$\forall x \forall y \forall z \quad (\texttt{Sold}(x,y,z) \wedge \texttt{Unregistered}(y)) \implies \texttt{Criminal}(x)$	△	$[\alpha_1, \beta_1]$
2	$\exists y \quad \texttt{Owns}(\text{Red}, y) \wedge \texttt{Unregistered}(y)$	△	$[\alpha_2, \beta_2]$
3	$\forall y \quad (\texttt{Owns}(\text{Red}, y) \wedge \texttt{Unregistered}(y)) \implies \texttt{Sold}(\text{Lefty}, y, \text{Red})$	△	$[\alpha_3, \beta_3]$
4	$\texttt{Owns}(\text{Red}, \text{WeaponX}) \wedge \texttt{Unregistered}(\text{WeaponX})$	EI, 2	$[\alpha_4, \beta_4] = [\alpha_2, \beta_2]$
5	$(\texttt{Owns}(\text{Red}, \text{WeaponX}) \wedge \texttt{Unregistered}(\text{WeaponX})) \implies \texttt{Sold}(\text{Lefty}, \text{WeaponX}, \text{Red})$	UI, 3	$[\alpha_5, \beta_5] = [\alpha_3, \beta_3]$
6	$\texttt{Sold}(\text{Lefty}, \text{WeaponX}, \text{Red})$	MP, 5, 4	$[\alpha_6, \beta_6]$, with: $\alpha_6 = \begin{cases} \alpha_3, & \text{if } \alpha_3 > 1-\beta_2; \\ 0, & \text{if } \alpha_3 = 1-\beta_2; \\ \text{no solution}, & \text{otherwise;} \end{cases}$ $\beta_6 = \begin{cases} \beta_3, & \text{if } \beta_3 \le 1-\alpha_2; \\ \text{no solution}, & \text{otherwise.} \end{cases}$
7	$\texttt{Unregistered}(\text{WeaponX})$	AE, 4	$[\alpha_7, \beta_7] = [\alpha_2, \beta_2]$
8	$(\texttt{Sold}(\text{Lefty}, \text{WeaponX}, \text{Red}) \wedge \texttt{Unregistered}(\text{WeaponX})) \implies \texttt{Criminal}(\text{Lefty})$	UI, 1	$[\alpha_8, \beta_8] = [\alpha_1, \beta_1]$
9	$\texttt{Sold}(\text{Lefty}, \text{WeaponX}, \text{Red}) \wedge \texttt{Unregistered}(\text{WeaponX})$		$= [\min(\alpha_2, \alpha_6), \min(\beta_2, \beta_6)]$
10	$\texttt{Criminal}(\text{Lefty})$	MP, 8, 9	$[\alpha_{10}, \beta_{10}]$, with: $\alpha_{10} = \begin{cases} \alpha_1, & \text{if } \alpha_1 > 1-\beta_9; \\ 0, & \text{if } \alpha_1 = 1-\beta_9; \\ \text{no solution}, & \text{otherwise;} \end{cases}$ $\beta_{10} = \begin{cases} \beta_1, & \text{if } \beta_1 \le 1-\alpha_9; \\ \text{no solution}, & \text{otherwise.} \end{cases}$

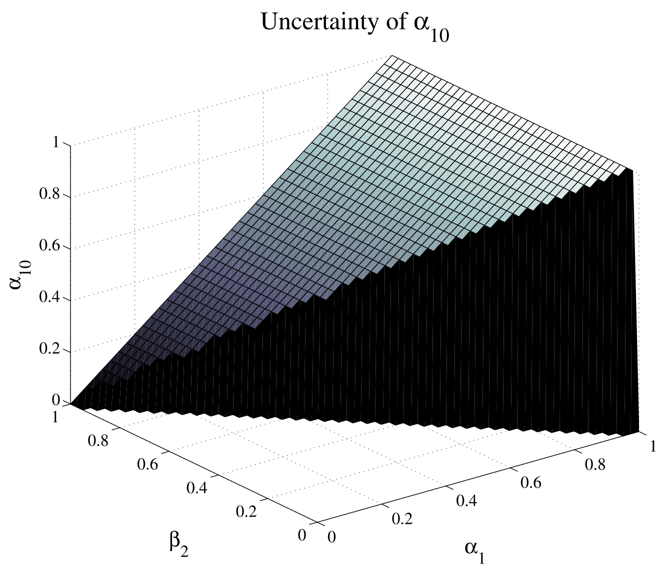

Fig. 1. Lower bound of the uncertainty of the conclusion `Criminal( Lefty )` (i.e., α_{10} of Table 3). The belief measure α_{10} increases as both the belief α_1 and the plausibility β_1 increase. Note how in the extreme case $\alpha_1 = \beta_1$ and $\alpha_1, \beta_1 \in \{0, 1\}$, the solution corresponds to that of classical logic. In addition, when $\alpha_1 = \beta_1$, the solution is a probabilistic model.

8. Satisfiability in ULP

The satisfiability (SAT) problem is a well known alternative for logic reasoning. Indeed, many automated reasoning problems in propositional logic are first reduced to satisfiability problems and then make use of a satisfiability solver.

Formally, the satisfiability problem consists in determining whether there exists a variable assignment such that every formula in a group of logic formulas (i.e., the model) evaluates to true [31]. This concept can be extended into ULP. In this case, instead of focusing on true/false variable assignments, we need to find the uncertainty intervals that satisfy a formula or a set of formulas. This is equivalent to finding a possible interpretation for the unknown uncertainties in a ULP model. Although this formulation departs from the traditional satisfiability formulation which identifies possible variable assignments, equivalent variable assignments in ULP can be found by identifying those grounded propositions with no uncertainty (i.e., with uncertainty interval $[1, 1]$).

It has been demonstrated that the SAT problem is NP complete. However, there are instances in many diverse areas for which this problem can be reformulated with simpler and efficient algorithms [31]. As it will become evident later in this section, this is also true for the case of ULP satisfiability, enabling scalable solutions for reasoning with ULP.

The satisfiability problem for ULP models can be formulated as an optimization problem, as follows.

8.1. ULP satisfiability as an optimization problem

Consider a set Φ of uncertain propositions containing:

$$\begin{aligned}
\{\ &\varphi_1(\cdot), \text{ with uncertainty } [\alpha_{\varphi_1}, \beta_{\varphi_1}],\\
&\varphi_2(\cdot), \text{ with uncertainty } [\alpha_{\varphi_2}, \beta_{\varphi_2}],\\
\ldots,\ &\varphi_l(\cdot), \text{ with uncertainty } [\alpha_{\varphi_l}, \beta_{\varphi_l}]\ \},
\end{aligned}$$

whose uncertainty intervals are known. Let us call these propositions the evidence. Also consider a set Ψ of uncertain propositions containing the elements:

$$\begin{aligned}
\{\ &\psi_1(\cdot), \text{ with uncertainty } [\alpha_{\psi_1}, \beta_{\psi_1}],\\
&\psi_2(\cdot), \text{ with uncertainty } [\alpha_{\psi_2}, \beta_{\psi_2}],\\
\ldots,\ &\psi_m(\cdot), \text{ with uncertainty } [\alpha_{\psi_m}, \beta_{\psi_m}]\ \},
\end{aligned}$$

whose uncertainty intervals are unknown. Both evidence and unknown propositions are components of a ULP model made of n logic expressions:

$$\begin{aligned}
&F_1 : f_1(\Phi, \Psi), \text{ with uncertainty } [\alpha_{F_1}, \beta_{F_1}],\\
&F_2 : f_2(\Phi, \Psi), \text{ with uncertainty } [\alpha_{F_2}, \beta_{F_2}], \quad \ldots,\\
&F_n : f_n(\Phi, \Psi), \text{ with uncertainty } [\alpha_{F_n}, \beta_{F_n}],
\end{aligned}$$

where $f_1, f_2, \ldots, f_n$ are logic formulas. Without loss of generality, assume that these formulas are disjunctions of a subset of propositions in Φ and Ψ. For example, a ULP model could be defined as:

$$\begin{aligned}
&F_1 : \varphi_1 \vee \psi_1, && \text{with uncertainty } [\alpha_{F_1}, \beta_{F_1}],\\
&F_2 : \varphi_2 \vee \varphi_3 \vee \psi_1, && \text{with uncertainty } [\alpha_{F_2}, \beta_{F_2}], \text{ and}\\
&F_3 : \varphi_3 \vee \psi_2, && \text{with uncertainty } [\alpha_{F_3}, \beta_{F_3}].
\end{aligned} \tag{28}$$

Furthermore, let us denote as $\alpha_{\Phi\in F_j}$ and $\alpha_{\Psi\in F_j}$ the set of beliefs (i.e., lower bound of the uncertainty intervals) of the propositions in Φ and Ψ, respectively, that are part of the formula F_j, $j = 1, 2, \ldots, n$. Similarly, let us denote as $\beta_{\Phi\in F_j}$ and $\beta_{\Psi\in F_j}$ the set of plausibilities (i.e., upper bound of the uncertainty intervals) of the propositions in Φ and Ψ, respectively, that are part of the formula F_j. For example, in the model defined by (28), the sets $\alpha_{\Phi\in F_j}$, $\beta_{\Phi\in F_j}$, $\alpha_{\Psi\in F_j}$, $\beta_{\Psi\in F_j}$, with $j = 1, 2, 3$, are:

$$\begin{aligned}
&\alpha_{\Phi\in F_1} = \{\alpha_{\varphi_1}\}, && \beta_{\Phi\in F_1} = \{\beta_{\varphi_1}\},\\
&\alpha_{\Phi\in F_2} = \{\alpha_{\varphi_2}, \alpha_{\varphi_3}\}, && \beta_{\Phi\in F_2} = \{\beta_{\varphi_2}, \beta_{\varphi_3}\},\\
&\alpha_{\Phi\in F_3} = \{\alpha_{\varphi_3}\}, && \beta_{\Phi\in F_3} = \{\beta_{\varphi_3}\},\\
&\alpha_{\Psi\in F_1} = \{\alpha_{\psi_1}\}, && \beta_{\Psi\in F_1} = \{\beta_{\psi_1}\},\\
&\alpha_{\Psi\in F_2} = \{\alpha_{\psi_1}\}, && \beta_{\Psi\in F_2} = \{\beta_{\psi_1}\},\\
&\alpha_{\Psi\in F_3} = \{\alpha_{\psi_2}\}, \quad \text{and} && \beta_{\Psi\in F_3} = \{\beta_{\psi_2}\}.
\end{aligned}$$

Then, the satisfiability problem in ULP can be defined as finding the uncertainty intervals $[\alpha_{\psi_i}, \beta_{\psi_i}]$ that solve the following optimization problem:

$$\underset{\{[\alpha_{\psi_i},\beta_{\psi_i}]\}}{\text{minimize}} \quad \sum_{j=1}^{n}(\alpha_{F_j}-\hat{\alpha}_{F_j})^2+(\beta_{F_j}-\hat{\beta}_{F_j})^2 \tag{29a}$$

subject to

for all $j \in \{1,2,\ldots,n\}$:

$$\hat{\alpha}_{F_j}=\max(\alpha_{\Phi\in F_j},\alpha_{\Psi\in F_j}); \tag{29b}$$

$$\hat{\beta}_{F_j}=\max(\beta_{\Phi\in F_j},\beta_{\Psi\in F_j}); \tag{29c}$$

$$0\le\hat{\alpha}_{F_j}\le\hat{\beta}_{F_j}\le 1; \text{ and} \tag{29d}$$

$$0\le\alpha_{\psi_i}\le\beta_{\psi_i}\le 1; i\in\{1,\ldots,m\}. \tag{29e}$$

It is important to note the following:

- The *cost function* in (29a) is based on an l_2 norm. However, the definition of the cost function should not be considered restricted to l_2 norms. Other norms could be used to enforce particular properties of the solution.
- The *constraints* (29b) and (29c) model the uncertainty of an OR operation in CFE-based ULP, as defined in Section 5.2 above. The use of a disjunctive model simplifies the formulation of the optimization problem. Similar constraints could be constructed to model logic formulas that involve other logic operators, or, alternatively, any other logic expression could be converted into a disjunctive model. Constraints (29d) and (29e) condition the uncertainty intervals to be defined as closed sets, consistent with DS theory definitions.
- When the solution of the optimization problem renders a cost function equal to zero, the logic model is satisfiable. In this case, there may be an infinite number of solutions to the optimization problem, and an additional step should be added to the reasoning process for enforcing minimal commitment in the output intervals (i.e., delivering the most conservative interval allocation).

When the cost function in the solution does not evaluate to zero, the model is not satisfiable. In this case it is possible, however, to identify particular formulas that are making the problem unsatisfiable (by identifying the non-zero components in the cost function), and either discard them from the model, or properly weight the evidence to ensure satisfiability.

Also note that the optimization problem (29) is nonlinear. This problem, however, can be converted into a convex optimization problem if in each of the logic expressions $F_1, F_2, \ldots, F_n$ there is *at most* one proposition whose uncertainty is unknown. Under this assumption, the optimization problem (29) can be reformulated as:

$$\underset{\{[\alpha_{\psi_i},\beta_{\psi_i}]\}}{\text{minimize}} \quad \sum_{j=1}^{n}(\alpha_{F_j}-\hat{\alpha}_{F_j})^2+(\beta_{F_j}-\hat{\beta}_{F_j})^2 \tag{30a}$$

subject to

for all $j \in \{1,2,\ldots,n\}$:

$$0\le\hat{\alpha}_{F_j}\le\hat{\beta}_{F_j}\le 1; \tag{30b}$$

$$\hat{\alpha}_{F_j}=\alpha_{\Psi\in F_j}; \quad \hat{\beta}_{F_j}=\beta_{\Psi\in F_j}; \tag{30c}$$

$$\hat{\alpha}_{F_k}\ge\alpha_{\Phi\in F_j}; \quad \hat{\beta}_{F_k}\ge\beta_{\Phi\in F_j}; \tag{30d}$$

$$0\le\alpha_{\psi_i}\le\beta_{\psi_i}\le 1; i\in\{1,\ldots,m\}. \tag{30e}$$

As a convex optimization problem, the complexity of this algorithm is significantly less than that of its corresponding nonlinear problem. This makes this approach valuable in scenarios with hundreds to thousands of logic formulas such as the tracking problem addressed in [32] and the following example.

8.2. Example: MLNs and optimization-based ULP

Consider the scenario described in [33], in which a knowledge base contains the following three rules:

1. "Friends of friends are friends";
2. "Smoking causes cancer"; and
3. "If two people are friends and one smokes, then so does the other".

As defined in [33], these rules could be expressed in first order logic as:

Table 4
Set of logic formulas (i.e., rules) and evidence for the example of Section 8.2. For simplicity, the duals of evidence expressions to E_1, E_2, E_3, and E_4 are not shown in the table. These duals enforce an assumption of symmetry of a friendship relation (i.e., for two subjects d_1, d_2, the uncertainty of Friends(d_1, d_2) is the same of Friends(d_2, d_1).

	Logic expression	Uncertainty
Knowledge base of uncertain formulas/rules		
F_1	$\forall x \forall y \forall z$ Friends$(x, y) \wedge$ Friends$(y, z) \implies$ Friends(x, z)	$[\alpha_{F1}, \beta_{F1}] = [1, 1]$
F_2	$\forall x$ Smokes$(x) \implies$ Cancer(x)	$[\alpha_{F2}, \beta_{F2}]$
F_3	$\forall x \forall y$ Friends$(x, y) \wedge$ Smokes$(x) \implies$ Smokes(y)	$[\alpha_{F3}, \beta_{F3}]$
Evidence		
E_1	Friends(Ivan, John)	$[\alpha_{E1}, \beta_{E1}] = [1, 1]$
E_2	Friends(Katherine, Lars)	$[\alpha_{E2}, \beta_{E2}] = [1, 1]$
E_3	Friends(Michael, Nick)	$[\alpha_{E3}, \beta_{E3}] = [1, 1]$
E_4	Friends(Ivan, Michael)	$[\alpha_{E4}, \beta_{E4}] = [1, 1]$
E_5	Smokes(Ivan)	$[\alpha_{E5}, \beta_{E5}]$
E_6	Smokes(Nick)	$[\alpha_{E6}, \beta_{E6}]$

1. $\forall x \forall y \forall z$ Friends$(x, y) \wedge$ Friends$(y, z) \implies$ Friends(x, z);
2. $\forall x$ Smokes$(x) \implies$ Cancer(x); and
3. $\forall x \forall y$ Friends$(x, y) \wedge$ Smokes$(x) \implies$ Smokes(y),

respectively. Note that these types of rules would be rarely completely true (or false) in real-life problems [33]. However, if we are able to quantify and attach uncertainty measures to these rules, then we would have a valuable set of rules able to support meaningful automated reasoning. When using probabilities as the uncertainty measure, problems like the one in this example can be modeled (and solved) using MLNs. They could also be solved using ULP to take advantage of the increased degree of freedom and logic-consistency properties introduced in this paper.

Let us enhance this set of rules with a set of uncertain logic expressions and rules, as shown in Table 4. This set of logic expressions is applied on a domain of people defined as $\Theta_p = \{\text{Ivan}, \text{John}, \text{Katherine}, \text{Lars}, \text{Michael}, \text{Nick}\}$. For ease of explanation, friendship relations have been assumed to be perfectly known (i.e., with no uncertainty), with uncertainty intervals $[1, 1]$. The knowledge base and set of evidence are propotitionalized (i.e., grounded) over the domain Θ_p, along with the grounded propositions Smokes(·) and Cancer(·). The propotitionalized knowledge base and evidence sets are further processed to convert them into conjunctive normal form, rendering a total of 438 propositions. Then, we automatically formulate the ULP convex optimization model (30) for this example and use it for answering questions regarding the uncertainty of particular logic propositions.

Let us consider first the effect that changing the uncertainty of rule F_3 has in the uncertainty of the proposition Smokes(). Fig. 2 shows the uncertainty of this proposition as it applies to Katherine when the uncertainty of F_3 changes. In the figure, nine uncertainty intervals are considered for F_3, namely $[\alpha_{F3}, \beta_{F3} \in \{[1.00, 1.00], [0.75, 1.00], [0.50, 1.00], [0.50, 0.75], [0.50, 0.50], [0.25, 0.50], [0.00, 0.50], [0.00, 0.25], [0.00, 0.00]\}$. These uncertainty intervals represent various uncertainty conditions, including F_3 being completely true to completely false, as well as the uncertainty of F_3 being modeled with both probabilistic and DS models. This figure also shows the corresponding probabilities computed with MLNs (using Alchemy 1.0 [34]). Since MLNs rules require setting a weight for each logic expression, we set the weights of each proposition using the pignistic probability of its corresponding uncertainty interval. We can observe the following:

- The uncertainty interval of the proposition Smokes(Katherine) resembles the uncertainty interval of rule F_3. This is expected given that, as described in Section 7 above, the uncertain model of implication rules (as the one in F_3) is bounded by the uncertainty of the rule.
- The pignistic probability (BetP curve in the figure) provides a probabilistic solution based on the ULP models.
- Unless there is complete certainty of the truthfulness of F_3, the probabilistic result rendered by an MLN model quickly drops to a value close to 0.4. Although for most of the intervals considered in the figure this probability value falls outside of the uncertainty interval provided by ULP, this result is the best an MLN can provide given its underlying probabilistic model. Recall that, unlike ULP, the probabilistic model of MLNs cannot naturally model the uncertainty of the negation of F_3. In addition, by definition, the MLN model does not ensure consistency with a classical logic model and there is no direct correspondence between a weight in a logic formula and a probability value. With these limitations, the best solution that the MLN can provide for the proposition F_3 is close to 0.5 (i.e., both events "Katherine smokes" and "Katherine does not smoke" are almost equally likely).

Let us now focus on the effect of changing the uncertainty of the rule F_2, and assume that F_3 is always true (with uncertainty $[1, 1]$). Fig. 3 shows the uncertainty of the proposition Cancer(Ivan) as a function of the uncertainty interval $[\alpha_{F2}, \beta_{F2}]$. As in the analysis of the rule F_3, the following nine uncertainty intervals are considered for F_3: $[\alpha_{F2}, \beta_{F2}] \in \{[1.00, 1.00], [0.75, 1.00], [0.50, 1.00], [0.50, 0.75], [0.50, 0.50], [0.25, 0.50], [0.00, 0.50], [0.00, 0.25], [0.00, 0.00]\}$. Note how, from interval $[1.00, 1.00]$ down to interval $[0.50, 0.50]$ the uncertainty of Cancer(Ivan) follows the uncertainty of the

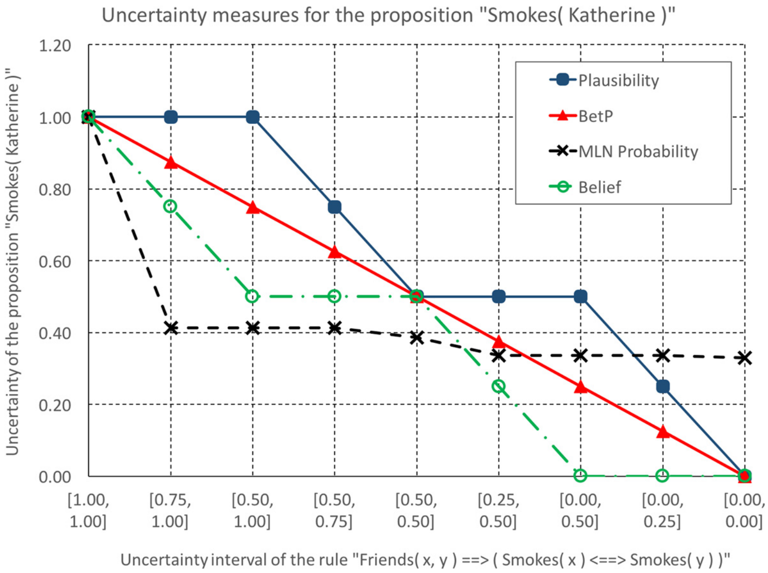

Fig. 2. Uncertainty of the proposition `Smokes(Kate)` as a function of the uncertainty interval $[\alpha_{F3}, \beta_{F3}]$, which indicates the uncertainty of the logic formula F_3 in the example of Section 8.2. The figure shows the belief and plausibility of `Smokes(Kate)` using ULP. In addition, the figure shows a mapping of the resulting ULP model into a probabilistic model using the pignistic transformation (labeled as BetP in the figure), as well as a probabilistic estimate rendered by an MLN model. Note how the uncertainty of `Smokes(Kate)` rendered by ULP follows the behavior of the uncertainty of F_3, and provides information even if the uncertainty of the rule indicates that F_3 is false.

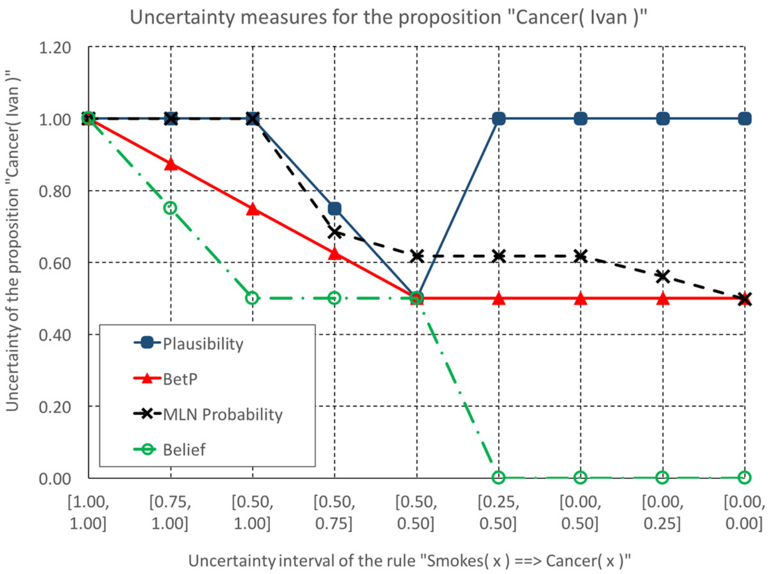

Fig. 3. Uncertainty of the proposition `Cancer(Ivan)` as a function of the uncertainty interval $[\alpha_{F2}, \beta_{F2}]$, which indicates the uncertainty of the logic formula F_2 in the example of Section 8.2. The figure shows the belief and plausibility of `Smokes(Kate)` using ULP. In addition, the figure shows a mapping of the resulting ULP model into a probabilistic model using the pianistic transformation (labeled as BetP in the figure), as well as a probabilistic estimate rendered by an MLN model. Note how the uncertainty of `Cancer(Ivan)` rendered by ULP follows the behavior of the uncertainty of F_2. When there is not enough evidence to support an assignment of an uncertainty interval of `Cancer(Ivan)`, the ULP model renders as a result the interval that represents complete ignorance, namely $[0, 1]$.

rule F_2. However, when there is insufficient evidence to support a satisfiable model a conclusion cannot be made on the uncertainty interval of `Cancer(Ivan)` in ULP. In this case, the uncertainty interval becomes $[0, 1]$ which indicates complete ignorance on the event `Cancer(Ivan)` being true or false. Note that rendering an uncertainty interval $[0, 1]$ in this case is consistent with the MP model (26) and (27): In the case of insufficient evidence to support a conclusion, the model is not satisfiable and a conservative solution $[0, 1]$ is output. Finally, the figure shows that the pignistic probabilistic transformation (i.e., BetP in the figure) renders results that are close to the output delivered by MLNs in this scenario.

9. Conclusion

We have presented the fundamental theory of *Uncertain Logic Processing* (ULP), a DS theoretic approach for first order logic operations. ULP provides support for handling uncertain fundamental logic operations (i.e., $\neg, \wedge, \vee, \implies$), combinations of these fundamental operations, variables, and logic quantifiers. The framework presented in this paper allows systematic generation of mass assignments based on uncertain first order logic formulas.

The use of DS theory as the substrate for modeling and tracking the propagation of uncertainty in the ULP framework provides us with important advantages such as the ability of modeling uncertainty using intervals (instead of being limited by sometimes restrictive probability values), ability for quantifying ignorance on the truth value of logical formulas or propositions, providing a principled way of refining uncertainty intervals without relying on priors or membership functions, and allowing compatibility with probabilistic models, among others.

ULP can be tuned for consistency with classical logic, rendering the classical logic results when the scenario represents "perfect" (i.e., without uncertainty) data/models. The consistency with classical logic gives the confidence to apply our proposed models as an extension of classical logic in reasoning. Given the parameterized presentation of the uncertain logic operations, extensions of ULP could be designed to address, in addition to consistency with classical logic, consistency with paraconsistent logics.

Acknowledgement

This work is based on research supported by the US Office of Naval Research (ONR) via grants #N00014-10-1-0140 and #N00014-11-1-0493, and the US National Science Foundation (NSF) via grants #1038257 and #1343430.

Appendix A. BBA for LC CFE-based AND

Consider the definition of the CFE fusion operator (see Section 2 above) and substitute the CFE coefficients for those of the AND operation, as indicated by Definition 11. Then: $m(d) = 2\gamma(d) + 2\gamma(\Theta)(\alpha_1 + \alpha_2)$.

- When $\delta_1 + \delta_2 \neq 0$:

$$\begin{aligned} m(d) &= \frac{\underline{\alpha}(\beta_1+\beta_2) - \underline{\beta}(\alpha_1+\alpha_2)}{\delta_1+\delta_2} + \frac{\underline{\delta}(\alpha_1+\alpha_2)}{\delta_1+\delta_2} \\ &= \tfrac{1}{\delta_1+\delta_2}(\underline{\alpha}\beta_1 + \underline{\alpha}\beta_2 - \alpha_1\underline{\beta} - \alpha_2\underline{\beta} + \underline{\delta}(\alpha_1+\alpha_2)). \end{aligned}$$

  Since $\underline{\delta} = \underline{\beta} - \underline{\alpha}$:

$$\begin{aligned} m(d) &= \tfrac{1}{\delta_1+\delta_2}(\underline{\alpha}\beta_1 + \underline{\alpha}\beta_2 - \alpha_1\underline{\beta} - \alpha_2\underline{\beta} \\ &\qquad + \alpha_1\underline{\beta} + \alpha_2\underline{\beta} - \alpha_1\underline{\alpha} - \alpha_2\underline{\alpha}) \\ &= \tfrac{1}{\delta_1+\delta_2}(\underline{\alpha}(\beta_2 - \alpha_2 + \beta_1 - \alpha_1)). \end{aligned} \tag{31}$$

  Substituting $\delta_1 = \beta_1 - \alpha_1$ and $\delta_2 = \beta_2 - \alpha_2$ in (31): $m(d) = \underline{\alpha}$.
- When $\delta_1 + \delta_2 = 0$, and making $\gamma(\Theta) = 0$: $m(d) = 2\gamma(d) = \underline{\alpha}$.

The mass $m(\overline{d})$ is given by:

$$m(\overline{d}) = \gamma_1(\overline{d}) + \gamma_1(\Theta)m_1(\overline{d}) + \gamma_2(\overline{d}) + \gamma_2(\Theta)m_2(\overline{d}). \tag{32}$$

Substituting the CFE coefficients as indicated by Definition 11 in (32): $m(\overline{d}) = 2\gamma(\overline{d}) + 2\gamma(\Theta)(2 - \beta_1 - \beta_2)$.

- When $\delta_1 + \delta_2 \neq 0$:

$$\begin{aligned} m(\overline{d}) &= \frac{\delta_1+\delta_2 - \underline{\beta}(2-\alpha_1-\alpha_2) + \underline{\alpha}(2-\beta_1-\beta_2)}{\delta_1+\delta_2} \\ &\quad + \frac{\underline{\delta}(2-\beta_1-\beta_2)}{\delta_1+\delta_2} \\ &= \tfrac{1}{\delta_1+\delta_2}(\delta_1+\delta_2 - \underline{\beta}(2-\alpha_1-\alpha_2) \\ &\qquad + \underline{\alpha}(2-\beta_1-\beta_2) + \underline{\delta}(2-\beta_1-\beta_2)). \end{aligned}$$

  Since $\underline{\delta} = \underline{\beta} - \underline{\alpha}$:

$$m(\overline{d}) = \frac{1}{\delta_1+\delta_2}(\delta_1 + \delta_2 - \underline{\beta}(2 - \alpha_1 - \alpha_2) + \underline{\alpha}(2 - \beta_1 - \beta_2) + (\underline{\beta} - \underline{\alpha})(2 - \beta_1 - \beta_2)) = \frac{1}{\delta_1+\delta_2}(\delta_1 + \delta_2 - \underline{\beta}(\beta_1 - \alpha_1 + \beta_2 - \alpha_2)). \quad (33)$$

Substituting $\delta_1 = \beta_1 - \alpha_1$ and $\delta_2 = \beta_2 - \alpha_2$ in (33): $m(\overline{d}) = 1 - \underline{\beta}$.

- When $\delta_1 + \delta_2 = 0$, and making $\gamma(\Theta) = 0$: $m(\overline{d}) = 2\gamma(\overline{d}) = 1 - \underline{\alpha} = 1 - \underline{\beta}$.

Finally, $m(\Theta) = 1 - m(d) - m(\overline{d}) = \underline{\beta} - \underline{\alpha}$.

Appendix B. Properties of the LC CFE-based UL operations

Consider logic expressions of the form $\varphi(d_i)$, with $1 \le i \le N$. Then, the following properties are satisfied:

1. *Idempotency*: This property is defined by: $\varphi_i(d) \wedge \varphi_i(d) = \varphi_i(d) \vee \varphi_i(d) = \varphi_i(d)$. In this case:

$$m_\wedge(d) = \min(\alpha_i, \alpha_i) = \alpha_i = \max(\alpha_i, \alpha_i) = m_\vee(d);$$
$$m_\wedge(\overline{d}) = 1 - \underline{\beta} = 1 - \min(\beta_i, \beta_i) = 1 - \beta_i = 1 - \max(\beta_i, \beta_i) = 1 - \overline{\beta} = m_\vee(\overline{d});$$
$$m_\wedge(\Theta) = \underline{\beta} - \underline{\alpha} = \beta_i - \alpha_i = \overline{\beta} - \overline{\alpha} = m_\vee(\Theta).$$

2. *Commutativity*: It refers to satisfying: $\varphi_1(d) \wedge \varphi_2(d) = \varphi_2(d) \wedge \varphi_1(d)$, and $\varphi_1(d) \vee \varphi_2(d) = \varphi_2(d) \vee \varphi_1(d)$. Let us call $m_{\varphi_i \wedge \varphi_j}(\cdot)$ the BBA resulting from $\varphi_i(d) \wedge \varphi_j(d)$, $i = \{1, 2\}$. Then, for the AND operation:

$$m_{\varphi_1\wedge\varphi_2}(d) = \min(\alpha_1, \alpha_2) = \min(\alpha_2, \alpha_1) = m_{\varphi_2\wedge\varphi_1}(d)$$
$$m_{\varphi_1\wedge\varphi_2}(\overline{d}) = 1 - \min(\beta_1, \beta_2) = m_{\varphi_2\wedge\varphi_1}(\overline{d})$$
$$m_{\varphi_1\wedge\varphi_2}(\Theta) = \min(\beta_1, \beta_2) - \min(\alpha_1, \alpha_2) = \min(\beta_2, \beta_1) - \min(\alpha_2, \alpha_1) = m_{\varphi_2\wedge\varphi_1}(\Theta).$$

A proof for commutativity for the logical OR operation is obtained by following a similar procedure.

3. *Associativity*: The associative property is defined by: $\varphi_1(d) \wedge [\varphi_2(d) \wedge \varphi_3(d)] = [\varphi_1(d) \wedge \varphi_2(d)] \wedge \varphi_3(d)$, and $\varphi_1(d) \vee [\varphi_2(d) \vee \varphi_3(d)] = [\varphi_1(d) \vee \varphi_2(d)] \vee \varphi_3(d)$. Let us call $\varphi_4(\cdot)$ the model generated by $\varphi_2(d) \wedge \varphi_3(d)$, and $\varphi_5(\cdot)$ the model generated by $\varphi_1(d) \wedge \varphi_2(d)$. Also, let us call $m_{\varphi_i \wedge \varphi_j}(\cdot)$ the BBA resulting from $\varphi_i(d) \wedge \varphi_j(d)$, $i = \{1, \dots, 5\}$. Our goal (for the AND) is to show that the model for $\varphi_1(\cdot) \wedge \varphi_4(\cdot)$ is equivalent to the model for $\varphi_5(\cdot) \wedge \varphi_3(\cdot)$:

$$m_{\varphi_1\wedge\varphi_4}(d) = \min(\alpha_1, \min(\alpha_2, \alpha_3)) = \min(\min(\alpha_1, \alpha_2), \alpha_3) = m_{\varphi_5\wedge\varphi_3}(d)$$
$$m_{\varphi_1\wedge\varphi_4}(\overline{d}) = 1 - \min(\beta_1, \min(\beta_2, \beta_3)) = 1 - \min(\min(\beta_1, \beta_2), \beta_3) = m_{\varphi_5\wedge\varphi_2}(\overline{d})$$
$$m_{\varphi_1\wedge\varphi_4}(\Theta) = \min(\min(\beta_1, \beta_2), \beta_3) - \min(\min(\alpha_1, \alpha_2), \alpha_3) = m_{\varphi_5\wedge\varphi_3}(\Theta).$$

A proof for associativity for the logical OR operation is obtained by following a similar procedure.

4. *Distributivity*: Distributive operations satisfy: $\varphi_1(d_i) \wedge [\varphi_2(d_j) \vee \varphi_3(d_k)] = [\varphi_1(d_i) \wedge \varphi_2(d_j)] \vee [\varphi_1(d_i) \wedge \varphi_3(d_j)]$, and $\varphi_1(d_i) \vee [\varphi_2(d_j) \wedge \varphi_3(d_k)] = [\varphi_1(d_i) \vee \varphi_2(d_j)] \wedge [\varphi_1(d_i) \vee \varphi_3(d_j)]$.
Let us call $\varphi_4(\cdot)$ the model generated by $\varphi_1(d) \wedge [\varphi_2(d) \vee \varphi_3(d)]$, and $\varphi_5(\cdot)$ the model generated by $[\varphi_1(d) \wedge \varphi_2(d)] \vee [\varphi_1(d) \wedge \varphi_3(d)]$. Our goal is to show that the model for $\varphi_4(\cdot)$ is equivalent to the model for $\varphi_5(\cdot)$. In general, these two models are:

$$m_{\varphi_4}(d) = \min(\alpha_1, \max(\alpha_2, \alpha_3));$$
$$m_{\varphi_4}(\overline{d}) = 1 - \min(\beta_1, \max(\beta_2, \beta_3));$$
$$m_{\varphi_4}(\Theta) = \min(\beta_1, \max(\beta_2, \beta_3)) - \min(\alpha_1, \max(\alpha_2, \alpha_3));$$

and

$$\begin{aligned}
m_{\varphi_5}(d) &= \max(\min(\alpha_1,\alpha_2),\min(\alpha_1,\alpha_3));\\
m_{\varphi_5}(\overline{d}) &= 1-\max(\min(\beta_1,\beta_2),\min(\beta_1,\beta_3));\\
m_{\varphi_5}(\Theta) &= \max(\min(\beta_1,\beta_2),\min(\beta_1,\beta_3))\\
&\quad -\max(\min(\alpha_1,\alpha_2),\min(\alpha_1,\alpha_3)).
\end{aligned}$$

Now, consider the focal set d. We have three cases (other possible cases are equivalent to these three after applying the commutativity rule): (a) $\alpha_1 \le \alpha_2 \le \alpha_3$; (b) $\alpha_2 \le \alpha_1 \le \alpha_3$; and (c) $\alpha_2 \le \alpha_3 \le \alpha_1$. The mass associated to the focal set d is: (a) $m_{\varphi_4}(d) = \alpha_1 = m_{\varphi_5}(d)$; (b) $m_{\varphi_4}(d) = \alpha_1 = m_{\varphi_5}(d)$; and (c) $m_{\varphi_4}(d) = \alpha_3 = m_{\varphi_5}(d)$; i.e., $m_{\varphi_4}(d) = m_{\varphi_5}(d)$ in all the cases. For the focal set $\overline{d}$ we also have three basic cases: (a) $\beta_1 \le \beta_2 \le \beta_3$; (b) $\beta_2 \le \beta_1 \le \beta_3$; and (c) $\beta_2 \le \beta_3 \le \beta_1$; which render: (a) $m_{\varphi_4}(\overline{d}) = 1-\beta_1 = m_{\varphi_5}(\overline{d})$; (b) $m_{\varphi_4}(\overline{d}) = 1-\beta_1 = m_{\varphi_5}(\overline{d})$; and (c) $m_{\varphi_4}(\overline{d}) = 1-\beta_3 = m_{\varphi_5}(\overline{d})$. Based on the cases above, it can be shown that also $m_{\varphi_4}(\Theta) = m_{\varphi_5}(\Theta)$, proving distributivity for the logical AND operation. A proof for distributivity for the logical OR operation is obtained by following a similar procedure.

Appendix C. ULP operators for multiple propositions

The CFE-based uncertain logic AND operator generalizes as follows for the general case of n propositions:

$$\begin{aligned}
&\bigwedge_{i=1}^{n}\varphi(d_i):\\
&m(\,\theta_{i^-}\times d_i\times\theta_{i^+}\,) \;=\; \tfrac{1}{n}\underline{\alpha}_{1:n}+\tfrac{1}{n}\left(\underline{\beta}_{1:n}-\underline{\alpha}_{1:n}\right)\alpha_i,\ \forall i;\\
&m(\,\theta_{i^-}\times \overline{d}_i\times\theta_{i^+}\,) \;=\; \tfrac{1}{n}\left(1-\underline{\beta}_{1:n}\right)\\
&\qquad\qquad +\tfrac{1}{n}\left(\underline{\beta}_{1:n}-\underline{\alpha}_{1:n}\right)(1-\beta_i),\ \forall i;\\
&m(\,\Theta_{\varphi,D}\,) \;=\; \tfrac{1}{n}\left(\underline{\beta}_{1:n}-\underline{\alpha}_{1:n}\right)\left(\textstyle\sum_{i=1}^{n}(\beta_i-\alpha_i)\right),
\end{aligned}\tag{34}$$

with: $\theta_{i^-} = \Theta_{\varphi,d_1}\times\Theta_{\varphi,d_2}\times\ldots\times\Theta_{\varphi,d_{i-1}}$; $\theta_{i^+} = \Theta_{\varphi,d_{i+1}}\times\ldots\times\Theta_{\varphi,d_{n-1}}\times\Theta_{\varphi,d_n}$; $\underline{\alpha}_{1:n} = \underline{\alpha}_{1,2,\ldots,n} = \min(\alpha_1,\alpha_2,\ldots,\alpha_N)$; and $\underline{\beta}_{1:n} = \underline{\beta}_{1,2,\ldots,n} = \min(\beta_1,\beta_2,\ldots,\beta_N)$.

Similarly, for the logical OR operator:

$$\begin{aligned}
&\bigvee_{i=1}^{n}\varphi(d_i):\\
&m(\,\theta_{i^-}\times d_i\times\theta_{i^+}\,) \;=\; \tfrac{1}{n}\overline{\alpha}_{1:n}+\tfrac{1}{n}\left(\overline{\beta}_{1:n}-\overline{\alpha}_{1:n}\right)\alpha_i,\ \forall i;\\
&m(\,\theta_{i^-}\times \overline{d}_i\times\theta_{i^+}\,) \;=\; \tfrac{1}{n}\left(1-\overline{\beta}_{1:n}\right)\\
&\qquad\qquad +\tfrac{1}{n}\left(\overline{\beta}_{1:n}-\overline{\alpha}_{1:n}\right)(1-\beta_i),\ \forall i;\\
&m(\,\Theta_{\varphi,D}\,) \;=\; \tfrac{1}{n}\left(\overline{\beta}_{1:n}-\overline{\alpha}_{1:n}\right)\left(\textstyle\sum_{i=1}^{n}(\beta_i-\alpha_i)\right),
\end{aligned}$$

with: $\overline{\alpha}_{1:n} = \overline{\alpha}_{1,2,\ldots,n} = \max(\alpha_1,\alpha_2,\ldots,\alpha_n)$ and $\overline{\beta}_{1:n} = \overline{\beta}_{1,2,\ldots,n} = \max(\beta_1,\beta_2,\ldots,\beta_n)$.

Simpler models are obtained for the expressions $\bigwedge_{i=1}^{M}\varphi_i(d)$ and $\bigvee_{i=1}^{M}\varphi_i(d)$:

$$\begin{aligned}
&\bigwedge_{i=1}^{M}\varphi_i(d): m(d) = \underline{\alpha}_{1:M}+\tfrac{1}{M}\left(\underline{\beta}_{1:M}-\underline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}\alpha_i\right);\\
&m(\overline{d}) = \left(1-\underline{\beta}_{1:M}\right)+\tfrac{1}{M}\left(\underline{\beta}_{1:M}-\underline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}(1-\beta_i)\right);\\
&m(\Theta_{\varphi_{1:M},d}) = \tfrac{1}{M}\left(\underline{\beta}_{1:M}-\underline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}(\beta_i-\alpha_i)\right),
\end{aligned}$$

and

$$\begin{aligned}
&\bigvee_{i=1}^{M}\varphi_i(d): m(d) = \overline{\alpha}_{1:M}+\tfrac{1}{M}\left(\overline{\beta}_{1:M}-\overline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}\alpha_i\right);\\
&m(\overline{d}) = \left(1-\overline{\beta}_{1:M}\right)+\tfrac{1}{M}\left(\overline{\beta}_{1:M}-\overline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}(1-\beta_i)\right);\\
&m(\Theta_{\varphi_{1:M},d}) = \tfrac{1}{M}\left(\overline{\beta}_{1:M}-\overline{\alpha}_{1:M}\right)\left(\textstyle\sum_{i=1}^{M}(\beta_i-\alpha_i)\right).
\end{aligned}$$

References

[1] N.J. Nilsson, Probabilistic logic, Artif. Intell. 28 (1) (1986) 71–88.
[2] L.A. Zadeh, Fuzzy sets, Inf. Control 8 (3) (1965) 338–353.
[3] M. Richardson, P. Domingos, Markov logic networks, Mach. Learn. 62 (1–2) (2006) 107–136.
[4] A. Kimmig, S.H. Bach, M. Broecheler, B. Huang, L. Getoor, A short introduction to probabilistic soft logic, in: NIPS Workshop on Probabilistic Programming: Foundations and Applications, 2012.
[5] S. Destercke, T. Burger, Toward an axiomatic definition of conflict between belief functions, IEEE Trans. Cybern. 43 (2) (2013) 585–596.
[6] D. Dubois, H. Prade, Possibilistic logic: a retrospective and prospective view, Fuzzy Sets Syst. 144 (1) (2004) 3–23.
[7] G. Malinowski, Many-Valued Logics, Oxford Logic Guides, Clarendon Press, 1993.
[8] D. Dubois, H. Prade, Possibility Theory: An Approach to Computerized Processing of Uncertainty (Traduction Revue et Augmentée de "Théorie des Possibilités"), Plenum Press, New York, 1988.
[9] R. Fagin, J.Y. Halpern, Uncertainty, belief, and probability, in: Proceedings of the 11th International Joint Conference on Artificial Intelligence, vol. 2, IJCAI'89, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1989, pp. 1161–1167.
[10] E. Ruspini, The Logical Foundations of Evidential Reasoning, December 1986, SRI Technical Note 408.
[11] G.M. Provan, A logic-based analysis of Dempster–Shafer theory, Int. J. Approx. Reason. 4 (5–6) (1990) 451–495.
[12] G.M. Provan, A logical interpretation of Dempster–Shafer theory, with application to visual recognition, in: Fifth Conference on Uncertainty in Artificial Intelligence, UAI1989, 1989.
[13] L. Cholvy, Towards a logical belief function theory, J. Appl. Log. 13 (4) (2015) 441–457.
[14] A. Saffiotti, A belief-function logic, in: Proceedings of the Tenth National Conference on Artificial Intelligence, AAAI'92, AAAI Press, 1992, pp. 642–647.
[15] A. Saffiotti, E. Umkehrer, Inference-driven construction of valuation systems from first-order clauses, IEEE Trans. Syst. Man Cybern. 24 (11) (1994) 1611–1624.
[16] Context-dependent combination of sensor information in Dempster–Shafer theory for BDI, Knowl. Inf. Syst. 51 (1) (2017) 259–285, https://doi.org/10.1007/s10115-016-0978-0.
[17] A.S. Rao, AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language, Springer, Berlin, Heidelberg, 1996, pp. 42–55.
[18] B. Ristic, P. Smets, Target identification using belief functions and implication rules, IEEE Trans. Aerosp. Electron. Syst. 41 (3) (2005) 1097–1103.
[19] A. Benavoli, L. Chisci, A. Farina, B. Ristic, Modelling uncertain implication rules in evidence theory, in: 11th International Conference on Information Fusion, 2008.
[20] R.C. Núñez, M. Scheutz, K. Premaratne, M.N. Murthi, Modeling uncertainty in first-order logic: a Dempster–Shafer theoretic approach, in: 8th International Symposium on Imprecise Probability: Theories and Applications, 2013.
[21] R.C. Núñez, R. Dabarera, M. Scheutz, G. Briggs, O. Bueno, K. Premaratne, M.N. Murthi, DS-based uncertain implication rules for inference and fusion applications, in: 16th International Conference on Information Fusion, FUSION 2013, 2013, pp. 1934–1941.
[22] J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.
[23] G. Shafer, A Mathematical Theory of Evidence, Princeton University Press, 1976.
[24] T. Wickramarathne, K. Premaratne, M.N. Murthi, Consensus-based credibility estimation of soft evidence for robust data fusion, in: Proceedings of the 2nd International Conference on Belief Functions, Compiègne, France, 9–11 May 2012, in: Adv. Soft Comput., vol. 164, Springer, 2012, pp. 301–309.
[25] K. Premaratne, M.N. Murthi, J. Zhang, M. Scheutz, P.H. Bauer, A Dempster–Shafer theoretic conditional approach to evidence updating for fusion of hard and soft data, in: 12th International Conference on Information Fusion, 2009, FUSION '09, 2009, pp. 2122–2129.
[26] R. Fagin, J.Y. Halpern, A new approach to updating beliefs, in: Uncertainty in Artificial Intelligence, Elsevier Science Publishers, 1991, pp. 347–374.
[27] E. Kulasekere, K. Premaratne, D. Dewasurendra, M.-L. Shyu, P. Bauer, Conditioning and updating evidence, Int. J. Approx. Reason. 36 (1) (2004) 75–108.
[28] M.R. Genesereth, N.J. Nilsson, Logical Foundations of Artificial Intelligence, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.
[29] E.A. Bender, Mathematical Methods in Artificial Intelligence, IEEE Computer Society Press, Los Alamitos, CA, USA, 1996.
[30] D. Dubois, H. Prade, A set-theoretic view of belief functions, in: R. Yager, L. Liu (Eds.), Classic Works of the Dempster–Shafer Theory of Belief Functions, in: Studies in Fuzziness and Soft Computing, vol. 219, Springer, Berlin/Heidelberg, 2008, pp. 375–410.
[31] L. Zhang, S. Malik, The quest for efficient boolean satisfiability solvers, in: E. Brinksma, K. Larsen (Eds.), Computer Aided Verification, in: Lecture Notes in Computer Science, vol. 2404, Springer, Berlin, Heidelberg, 2002, pp. 17–36.
[32] R.C. Núñez, M.N. Murthi, K. Premaratne, Efficient computation of DS-based uncertain logic operations and its application to hard and soft data fusion, in: 17th International Conference on Information Fusion (FUSION 2014), 2014, pp. 1–8.
[33] P. Domingos, D. Lowd, Markov Logic: An Interface Layer for Artificial Intelligence, 1st edition, Morgan and Claypool Publishers, 2009.
[34] S. Kok, P. Singla, M. Richardson, P. Domingos, M. Sumner, H. Poon, D. Lowd, J. Wang, A. Nath, The Alchemy System for Statistical Relational AI: User Manual, Department of Computer Science and Engineering, University of Washington, January 2010.