



**QUEEN'S
UNIVERSITY
BELFAST**

A Syntactic Approach to Revising Epistemic States with Uncertain Inputs

Bauters, K., Liu, W., Hong, J., Godo, L., & Sierra, C. (2014). A Syntactic Approach to Revising Epistemic States with Uncertain Inputs. In *Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on* (pp. 154-161). Institute of Electrical and Electronics Engineers (IEEE). <https://doi.org/10.1109/ICTAI.2014.32>

Published in:

Tools with Artificial Intelligence (ICTAI), 2014 IEEE 26th International Conference on

Document Version:

Early version, also known as pre-print

Queen's University Belfast - Research Portal:

[Link to publication record in Queen's University Belfast Research Portal](#)

General rights

Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

A Syntactic Approach to Revising Epistemic States with Uncertain Inputs

Kim Bauters*, Weiru Liu*, Jun Hong*, Lluís Godo*[†] and Carles Sierra*[†]

*Queen’s University Belfast (QUB), Belfast, United Kingdom

Email: {k.bauters, w.liu, j.hong}@qub.ac.uk

[†]IIIA, CSIC, Bellaterra, Spain

Email: {godo, sierra}@iiia.csic.es

Abstract—Revising its beliefs when receiving new information is an important ability of any intelligent system. However, in realistic settings the new input is not always certain. A compelling way of dealing with uncertain input in an agent-based setting is to treat it as unreliable input, which may strengthen or weaken the beliefs of the agent. Recent work focused on the postulates associated with this form of belief change and on finding semantical operators that satisfy these postulates. In this paper we propose a new syntactic approach for this form of belief change and show that it agrees with the semantical definition. This makes it feasible to develop complex agent systems capable of efficiently dealing with unreliable input in a semantically meaningful way. Additionally, we show that imposing restrictions on the input and the beliefs that are entailed allows us to devise a tractable approach suitable for resource-bounded agents or agents where reactivity is of paramount importance.

I. INTRODUCTION

Revision plays an important role in intelligent systems, such as BDI agents [16], where we need to deal with the dynamic nature of knowledge and beliefs. In the seminal work [1] the authors introduced the so-called AGM postulates, which identify how new information should be added to a knowledge base to preserve consistency. Later, in [8], it was demonstrated that the AGM postulates were insufficient to deal with iterated belief revision. To overcome this problem four additional postulates were introduced that more precisely define how a revision operator should behave when dealing with iterated revision. A key concept was the realisation that the belief sets, as used in the AGM approach, are inadequate. Instead, belief revision should be a process on the epistemic state rather than the belief set (where the first induces the latter, but not vice versa). A common way of representing an epistemic state is by assigning weights to the various possible worlds. A natural extension to this idea is then to also associate a weight with the input resulting in weighted or uncertain input.

An important consideration when dealing with uncertain input is how the input should be interpreted [10]. One interpretation is that the input acts as a constraint that must be satisfied by the revision, i.e. after revision we want to be able to entail the given formula with exactly the specified degree of uncertainty. Another interpretation is that the uncertain input is treated as a new belief with an associated strength. The input is then seen as an extra piece of evidence that may or may

not be useful. This last approach is common in our everyday lives. Indeed, assume we have tickets for a sporting event and a single random person on social media tells you that the event is cancelled. Most likely, this single post will not be enough to convince you that the event is actually cancelled. However, as more and more people post messages that the event is cancelled you may eventually revise your opinion, depending on how ingrained the original belief is. Similarly, interpreting uncertain beliefs as new information with an associated strength makes sense in an agent-based setting where it should be the strength of the information that determines if it is adopted or not.

A framework for managing unreliable inputs, i.e. interpreted as new pieces of information that may or may not be relevant, has been presented in [14]. The authors of this work start by presenting a new definition of epistemic states that can be used to deal with different kinds of uncertain belief representations. In particular, their definition can be instantiated to, among others, ordinal conditional functions (OCF) [14] and representations based on infinitesimal probabilities [7]. Using this new and general definition of an epistemic state, the authors propose rational postulates that a belief change operator dealing with unreliable input should adhere to, along with a (semantical) belief change operator that satisfies these postulates. Furthermore, they show that their belief change operator reduces to iterated belief revision in the sense of Darwiche and Pearl [8] when new input has a higher strength than any of the current beliefs held by the agent. However, their work only focuses on the semantics and theoretical properties and does not consider practical approaches to belief change.

In this paper we introduce a syntactic operator for dealing with unreliable input. We prove that this syntactic operator corresponds to the semantic operator introduced in [14]. In addition, we explore the feasibility of a tractable approach. We show that tractability can be achieved by restricting the form of both the input and the language of formulas to be evaluated. Even though the input language is restricted, it remains highly expressive (e.g. formulas in DNF are allowed).

The remainder of the paper is organised as follows. Some preliminary notions are mentioned in Section II. A syntactic approach for dealing with uncertain input is described in Section III and a tractable approach is discussed in Section IV. Related work is discussed in Section V and conclusions are drawn in Section VI.

II. PRELIMINARIES

We start with some necessary preliminaries on epistemic states, which can be used to model the beliefs of an intelligent system or agent (Section II-A). Based on the beliefs held by an agent, we can verify if a given arbitrary formula is entailed, i.e. if it is believed. Commonly, such formulas are classical (i.e. Boolean) and entailment is only checked against the most strongly held beliefs. However, as shown in [3], it is possible to consider a more expressive language that also allows to reflect on the uncertainty associated with those beliefs (Section II-B).

A. Modelling Uncertain Beliefs as Epistemic States

To define epistemic states, we start with a finite set of atoms \mathcal{At} and a language \mathcal{L} constructed over it. We use \mathcal{Lit} to denote the set of literals that can be constructed from \mathcal{At} , i.e. $\mathcal{Lit} = \{a \mid a \in \mathcal{At}\} \cup \{\neg a \mid a \in \mathcal{At}\}$. For a literal $l \in \mathcal{Lit}$ we use l^* to denote the underlying atom, i.e. $l^* = a$ when $l = a$ or $l = \neg a$. Propositions in \mathcal{L} are defined in Backus-Naur Form (BNF) as $\varphi ::= a \mid \neg a \mid (\varphi_1 \wedge \varphi_2) \mid (\varphi_1 \vee \varphi_2)$, i.e. all propositions are in Negation Normal Form (NNF), where any arbitrary formula can be efficiently converted into NNF. We use $\text{lit}(\varphi)$ to denote the set of unique literals used in a formula φ . A possible world ω (or interpretation) is a function that maps \mathcal{At} onto $\{0, 1\}$. The set of all possible worlds is denoted by Ω , i.e. $\Omega = 2^{\mathcal{At}}$. A possible world ω is a model of a proposition φ iff $\omega(\varphi) = 1$, denoted as $\omega \models \varphi$. We use $\text{Mod}(\varphi)$ to denote the set of models of φ . An epistemic state, used to represent the beliefs of an agent, is then defined as:

Definition 1. (from [14]) Let Ω be a set of possible worlds. An *epistemic state* Φ is a mapping $\Phi : \Omega \rightarrow \mathbb{Z} \cup \{-\infty, +\infty\}$.

An epistemic state will be used to represent the mental state of an agent, where the value $\Phi(\omega)$ associated with a possible world ω , called the weight of ω , is understood as the degree of belief in the possible world ω . Throughout the paper we denote epistemic states using capital Greek letters. When $\Phi(\omega) = \infty$ (resp. $-\infty$) it means the agent believes ω is fully plausible (resp. not plausible at all) while $\Phi(\omega) = 0$ denotes that the agent is totally ignorant about ω . For $\omega, \omega' \in \Omega$ and $\Phi(\omega) > \Phi(\omega')$ the intuition is that ω is more plausible than ω' . When the agent learns the new information that φ holds, with weight m , then this is represented as a *simple epistemic state* Φ_{in} such that $\Phi_{in}(\omega) = m$ when $\omega \models \varphi$ and $\Phi_{in}(\omega) = 0$ otherwise. This simple epistemic state, which corresponds to the input (φ, m) , is then used to revise the agent's epistemic state Φ . The operator introduced in [14] to revise an epistemic state Φ by Φ' , denoted as $\Phi \circ \Phi'$, is $\forall \omega \in \Omega, (\Phi \circ \Phi')(\omega) = \Phi(\omega) + \Phi'(\omega)$. Since an input corresponds to a simple epistemic state, we often simply write $\Phi \circ (\varphi, m)$. We will also use $\Phi \circ I$ with $I = \langle i_1, \dots, i_n \rangle$ a sequence of inputs to denote $\Phi \circ i_1 \circ \dots \circ i_n$.

It is important to clarify that the definition of an epistemic state given in Definition 1 allows for the construction of a general framework for dealing with uncertain beliefs. Indeed, this definition does not impose any restrictions on the values associated with the possible worlds, other than that they are weights. Other representations for epistemic states, which

attach more specific meaning to the values, have been shown to be equivalent to the one from Definition 1. Specifically, Definition 1 induces an Ordinal Conditional Function (OCF) [17], [14]¹, which in turn can be transformed into other representations, e.g. those based on infinitesimal probabilities [7] and possibility theory [11]. The representation from Definition 1 can thus be *instantiated* using any of the other representations to best suit the nature of the uncertainty. The epistemic state given in Definition 1, however, is easier to work with as it relies on integers (and not e.g. ordinal numbers) and because it does not need a normalisation step (e.g. as needed in OCF).

Example 1. Let $\mathcal{At} = \{a, b, c\}$. Consider the epistemic state Φ such that $\Phi(\{a, b, c\}) = \Phi(\{a, \neg b, c\}) = \Phi(\{a, b, \neg c\}) = \Phi(\{a, \neg b, \neg c\}) = 3$ and $\Phi(\omega) = 0$ for all other possible worlds ω . Intuitively, this models an agent that believes 'a' is more plausible than ' $\neg a$ '. Indeed, exactly those possible worlds that model 'a' have a higher weight than the others. We say that the agent believes 'a' with a strength of 3 and is ignorant about the other literals in \mathcal{Lit} . Now consider the input $(c, 2)$. This input corresponds to the simple epistemic state Φ' for which $\Phi'(\{a, b, c\}) = \Phi'(\{a, \neg b, c\}) = \Phi'(\{\neg a, b, c\}) = \Phi'(\{\neg a, \neg b, c\}) = 2$ while for all other worlds ω we have that $\Phi'(\omega) = 0$. The result of revising Φ given the input, denoted as $\Psi = \Phi \circ (c, 2)$, is given by Ψ such that:

$$\begin{aligned} \Psi(\{a, b, c\}) &= 5 & \Psi(\{\neg a, b, c\}) &= 2 \\ \Psi(\{a, \neg b, c\}) &= 5 & \Psi(\{\neg a, \neg b, c\}) &= 2 \\ \Psi(\{a, b, \neg c\}) &= 3 & \Psi(\{\neg a, b, \neg c\}) &= 0 \\ \Psi(\{a, \neg b, \neg c\}) &= 3 & \Psi(\{\neg a, \neg b, \neg c\}) &= 0 \end{aligned}$$

In other words: the agent most strongly believes that both 'a' and 'c' are true in the real world, as expected, while still being ignorant as to whether 'b' is true or false.

The belief set, i.e. the sentences that an agent is committed to believe, is commonly defined as the set that has all the most plausible worlds as its models. To define this set, we first define the notion of a preorder. A *preorder* \leq_A on a set A is a reflexive and transitive relation over $A \times A$. We say that \leq_A is *total* iff for all $a, b \in A$ we have that either $a \leq_A b$ or $b \leq_A a$. *Definition 2.* (from [14]) Let Φ be an epistemic state. We have that $\text{Bel}(\Phi) = \{\varphi \in \mathcal{L} \mid \omega \models \varphi \text{ for all } \omega \in \min(\Omega, \leq_\Phi)\}$ is the belief set of Φ . Here \leq_Φ is a total preorder relation over Ω such that $\omega \leq_\Phi \omega'$ iff $\Phi(\omega) \geq \Phi(\omega')$ and $\min(\Omega, \leq_\Phi)$ denotes the set of minimal elements of Ω according to \leq_Φ .

We can also specify $\text{Bel}(\Phi)$ as the strongest (i.e. having the least models) proposition φ such that $\text{Mod}(\varphi) = \min(\Omega, \leq_\Phi)$. This proposition is, of course, unique up to logical equivalence.

Example 2. Consider Ψ from Example 1. The models with the highest weight are given by $\min(\Omega, \leq_\Psi) = \{\{a, b, c\}, \{a, \neg b, c\}\}$ and thus $\text{Bel}(\Psi) = a \wedge c$. We can easily verify that the agent believes that 'a' must be true, since $a \wedge c \models a$. Similarly, the agent does not believe that 'b' must also be true since $a \wedge c \not\models a \wedge b$. This is as expected, given that the agent is ignorant about the actual truth value of 'b'.

¹In addition, in [14] it has been shown that revising an epistemic state with an uncertain input is equivalent to combining the two corresponding OCFs using the combination operator suggested in [13].

B. Reasoning about Uncertain Beliefs

A belief set only gives us information about the most plausible beliefs. Still, an epistemic state also contains information on whether one option is more plausible than another, even if we currently believe neither to be true [3]. To deal with this, the language \mathcal{L}_{\geq} , which extends \mathcal{L} , can be defined in BNF as:

$$\varphi ::= a \mid \neg a \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \geq \varphi_2 \mid \varphi_1 > \varphi_2$$

To define the semantics of \mathcal{L}_{\geq} , we first define a mapping λ , which maps arbitrary formulas $\varphi \in \mathcal{L}_{\geq}$ onto $\mathbb{Z} \cup \{-\infty, +\infty\}$:

$$\lambda(\varphi) = \begin{cases} \max\{\Phi(\omega) \mid \omega \models \varphi\} & \text{if } \varphi \in \mathcal{L} \\ \lambda(\text{pare}(\varphi)) & \text{otherwise} \end{cases}$$

with $\max(\emptyset) = -\infty$. The intuition of $\lambda(\varphi)$ is that it is the maximum weight that can be associated with the formula φ , i.e. it reflects how strongly the agent believes φ to be true. When φ is not a propositional statement (i.e. $\varphi \notin \mathcal{L}$), we need to pare down the formula until the formula is a classical propositional statement. For this we define the function $\text{check}(\varphi)$ which determines if φ needs to be further pared down, i.e. $\text{check}(\varphi) = \varphi$ if $\varphi \in \mathcal{L}$ and otherwise $\text{check}(\varphi) = \text{pare}(\varphi)$. We then have $\text{pare}(\varphi \otimes \psi) = \text{check}(\varphi) \otimes \text{check}(\psi)$ with $\otimes \in \{\wedge, \vee\}$ and

$$\text{pare}(\varphi \geq \psi) = \begin{cases} \top & \text{if } \lambda(\neg\varphi) \leq \lambda(\neg\psi) \\ \perp & \text{otherwise} \end{cases}$$

In other words: \geq (or $>$, which is defined equivalently) is treated as a plausibility ordering where an expression such as $\varphi > \psi$ is read as “ φ is more plausible than ψ ” or, “we have less reason to believe $\neg\varphi$ than $\neg\psi$ ”. Such an expression can always be evaluated to true or false, i.e. \top (tautology) or \perp (contradiction).

Definition 3. Let Φ be an epistemic state and φ a formula in \mathcal{L}_{\geq} . We say that φ is entailed by Φ , written as $\Phi \models \varphi$, if and only if $\lambda(\varphi) > \lambda(\neg\varphi)$.

Example 3. Consider Φ' from Example 1. We have that $\Phi' \models (a \wedge c) \wedge (b > \neg c)$. Indeed, we have that $\lambda((a \wedge c) \wedge (b > \neg c)) > \lambda((\neg a \vee \neg c) \vee (\neg c \geq b))$. We need to pare down the formula on the left-hand side for which we find $\lambda(b) > \lambda(\neg c)$ since $\lambda(b) = 5$ and $\lambda(\neg c) = 3$, i.e. we find $\lambda(a \wedge c \wedge \top) = 5$. Similarly, for the right-hand side, we find a λ -value of 3. We can thus derive that the agent not only believes that $a \wedge c$ is true (in the sense of beliefs in classical logic), but also that the agent believes b to be more plausible than $\neg c$.

Any proposition in \mathcal{L}_{\geq} can also be turned into an equivalent formula in NNF form in the usual way and by rewriting $\neg(\psi_1 \geq \psi_2)$ as $(\psi_2 > \psi_1)$ and $\neg(\psi_1 > \psi_2)$ as $(\psi_2 \geq \psi_1)$. We assume this has been done when needed throughout paper.

III. SYNTACTIC REVISION

In this section we develop a general syntactic approach to belief change with uncertain inputs, suitable for use in a BDI setting. In particular, since a BDI agent not only has to revise its beliefs but even more frequently has to

verify the applicability of plans, we balance the complexity of belief change and belief entailment accordingly. In addition, as previously mentioned, we want the context of a plan to be formulas in the language \mathcal{L}_{\geq} so that the agent can reflect upon its own uncertain beliefs.

First, we need to decide on a syntactic representation of the beliefs currently held by an agent. A common way of syntactically representing an epistemic state Φ is by means of a finite set of weighted formulas (ψ, m) with m the weight of formula ψ (e.g. [18], [4]). However, such a representation would make it complex to verify if a belief is entailed when considering uncertain input. Indeed, the weight of a possible world ω in the semantical representation might be dependent on the weight of some or all of the formulas in the syntactic representation. Instead, we propose a syntactic representation that is closer to the semantical one by requiring that no two weighted formulas are pairwise satisfiable:

Definition 4. A *weighted belief base* \mathcal{B} is a set of formulas of the form (ψ, m) with $\psi \in \mathcal{L}$ and $m \in \mathbb{Z}$ so that there does not exist $(\psi_i, m_i), (\psi_j, m_j) \in \mathcal{B}$ for which $\psi_i \wedge \psi_j$ is satisfiable.

On a semantical level, this definition enforces that $\text{Mod}(\psi_i) \cap \text{Mod}(\psi_j) = \emptyset$, i.e. none of the classical formulas in \mathcal{B} have models in common. While this representation might at first appear restrictive, it is important to note that on the semantical level the possible worlds adhere to this exact same restriction. Intuitively, in a weighted belief base \mathcal{B} , each formula $(\psi, m) \in \mathcal{B}$ will correspond to the set of possible worlds $\text{Mod}(\psi)$ which all share the same weight.

Example 4. Consider the epistemic states Φ and Φ' from Example 1. We can compactly represent Φ using the weighted belief base $\{(a, 3)\}$. Similarly, we can represent Φ' using the weighted belief base $\{(a \wedge c, 5), (\neg a \wedge c, 2), (a \wedge \neg c, 3)\}$.

Next, we formalise the intuition from the previous example and define how the semantical representation corresponding with a weighted belief base \mathcal{B} can be retrieved.

Definition 5. Let \mathcal{B} be a weighted belief base. The epistemic state $\Phi_{\mathcal{B}}$ defined as

$$\Phi_{\mathcal{B}}(\omega) = \begin{cases} m & \text{if there exists a } (\psi, m) \in \mathcal{B} \text{ such that } \omega \models \psi \\ 0 & \text{otherwise} \end{cases}$$

is the (semantical) epistemic state induced by \mathcal{B} .

The above definition formalises the intuition that every formula (φ, m) in \mathcal{B} corresponds to a set of models $\text{Mod}(\varphi)$ such that for every $\omega \in \text{Mod}(\varphi)$ we have that $\Phi_{\mathcal{B}}(\omega) = m$. Hence, as desired, a weighted belief base is a compact representation of a semantical epistemic state. Furthermore, every epistemic state can be represented as a compact weighted belief base. Indeed, for $\omega \in \Omega$ we can define the equivalence class $[\omega] = \{\omega' \in \Omega \mid \Phi(\omega) = \Phi(\omega')\}$, i.e. all possible worlds with the same weight. For each equivalence class $[\omega]$ of Φ we then have that $(\varphi_{\omega}, \Phi(\omega)) \in \mathcal{B}$ with φ_{ω} a proposition such that $\text{Mod}(\varphi_{\omega}) = [\omega]$. So, φ_{ω} is a proposition that has as its models exactly those possible worlds that are in the equivalence class $[\omega]$. It then readily follows from Definition 5 that $\Phi = \Phi_{\mathcal{B}}$.

Importantly, a weighted belief base makes it straightforward to determine the weight associated with any arbitrary formula:

Proposition 1. *Let \mathcal{B} be a weighted belief base and $\varphi \in \mathcal{L}$ a formula. We have that $\lambda(\varphi) = m_i \neq 0$ iff there exists a $(\psi_i, m_i) \in \mathcal{B}$ such that $\psi_i \wedge \varphi$ is satisfiable and there does not exist a $(\psi_j, m_j) \in \mathcal{B}$ with $m_j > m_i$ such that $\psi_j \wedge \varphi$ is satisfiable. Otherwise, $\lambda(\varphi) = 0$.*

Proof: This readily follows from the definition of $\lambda(\varphi) = \max_{\omega \models \varphi} \Phi_{\mathcal{B}}(\omega)$ and from Definition 5. Indeed, since every formula in \mathcal{B} is a compact representation of a set of possible worlds ω and since none of the classical formulas in \mathcal{B} share models, the definition of $\lambda(\varphi)$ reduces to finding the formula (ψ_i, m_i) in \mathcal{B} with the highest weight m_i such that it shares models with φ , i.e. such that $\omega \models \varphi$ or $\varphi \wedge \psi_i$ is satisfiable. Otherwise, from Definition 5, we know that $\lambda(\varphi) = 0$. ■

Proposition 2. *Let \mathcal{B} be a weighted belief base, $\varphi \in \mathcal{L}$ a formula and $m_i \in \mathbb{Z}$ a weight. Determining whether $\lambda(\varphi) = m_i$ is NP-complete.*

Proof: It readily follows that this decision problem is in NP due to Proposition 1, i.e. verifying whether $\lambda(\varphi) = m_i$ requires $|\mathcal{B}|$ satisfiability checks, which is an NP-complete problem. To prove NP-hardness, we reduce the satisfiability problem, i.e. verifying whether a given formula ψ is satisfiable, to the problem of verifying whether $\lambda(\varphi) = m_i$. Without loss of generality, we can assume that ψ is in NNF. Let $\mathcal{B} = \{(\psi, 1)\}$ and $\varphi = a^\dagger \vee \neg a^\dagger$ with a^\dagger a fresh atom, i.e. φ is a tautology. We have that ψ is satisfiable iff $\lambda(\varphi) = 1$. Indeed, we know from Proposition 1 that determining $\lambda(\varphi) = 1$ is equivalent to verifying that $\psi \wedge \varphi$ is satisfiable or, equivalently, that ψ is satisfiable since φ is a tautology. ■

We now need to verify whether a syntactic revision operator actually exists that can transform a weighted belief base into another set of formulas given an arbitrary input. This new set of formulas then has to:

- (1) agree with the definition of a weighted belief base; and
- (2) correspond with belief change on the semantical level.

Before we define the syntactic revision operator, we first introduce some new terminology that will help us to simplify the definition. For a weighted belief base \mathcal{B} we use \mathcal{B}^* to denote the classical set of formulas, i.e. $\mathcal{B}^* = \{\psi_i \mid (\psi_i, m_i) \in \mathcal{B}\}$. The notation $\max(\mathcal{B})$ is used to denote the set of classical formulas in \mathcal{B} with the highest weight. In other words, we have that $\max(\mathcal{B}) = \{\psi_i \mid (\psi_i, m_i) \in \mathcal{B}, \nexists (\psi_j, m_j) \in \mathcal{B} \cdot m_j > m_i\}$. It follows from $\max(\mathcal{B})$ and Definition 5 that $\text{Bel}(\Phi_{\mathcal{B}}) = \bigvee \{\varphi \mid \varphi \in \max(\mathcal{B})\}$. Indeed, the models corresponding with the formulas in $\max(\mathcal{B})$ are exactly the models of the belief set of $\Phi_{\mathcal{B}}$. We can then define the syntactic operator as follows:

Definition 6. Let \mathcal{B} be a weighted belief base and (φ, μ) a non-trivial input, i.e. $\mu \neq 0$. Let $\mathcal{B}_\varphi = \{(\psi, m) \mid (\psi, m) \in \mathcal{B} \text{ and } \psi \wedge \varphi \text{ is consistent}\}$. We define \mathcal{B}' as:

$$\mathcal{B}' = \{(\psi \wedge \varphi, m + \mu), (\psi \wedge \neg \varphi, m) \mid (\psi, m) \in \mathcal{B}_\varphi\} \\ \cup \mathcal{B} \setminus \mathcal{B}_\varphi$$

The syntactic revision of \mathcal{B} with the input (φ, μ) , denoted as $\mathcal{B} \circ_s(\varphi, \mu)$, is given by $\mathcal{B} \circ_s(\varphi, \mu) = \mathcal{B}' \cup ((\varphi \wedge \neg \bigvee(\mathcal{B}_\varphi^*)), \mu)$.

Intuitively, $(\psi \wedge \varphi)$ describes the models shared by ψ and φ , $(\psi \wedge \neg \varphi)$ are those models in ψ but not in φ and $(\varphi \wedge \neg \bigvee(\mathcal{B}_\varphi^*))$

are those models in φ that are not yet in \mathcal{B}_φ^* . Also, whenever $\psi \wedge \neg \varphi$ is inconsistent we could simply omit it from \mathcal{B}' as it conveys no information.

Example 5. Let $\mathcal{B}_0 = \{\}$ be an empty weighted belief base. We revise \mathcal{B}_0 with $(a, 3)$ and then with $(c, 2)$. We trivially have that $\mathcal{B}_1 = \mathcal{B}_0 \circ_s(a, 3) = \{(a, 3)\}$ since \mathcal{B}_0 is empty. We find $\mathcal{B}_2 = \mathcal{B}_1 \circ_s(c, 2) = \{(a \wedge c, 5), (a \wedge \neg c, 3), (c \wedge \neg a, 2)\}$. We can check that $\max(\mathcal{B}_2) = \{a \wedge c\}$, i.e. we have that $\text{Bel}(\mathcal{B}_2) = a \wedge c$. Notice that the inputs coincide with Example 1, as does the resulting belief set. Furthermore, the results coincide with the intuition expressed in Example 4.

It is easy to see that the syntactic operator only grows \mathcal{B} in polynomial space. Indeed, after each revision with an input the number of new weighted formulas is bounded by $\mathcal{O}(2 \cdot |\mathcal{B}| + 1)$ with $|\mathcal{B}|$ the number of weighted formulas in \mathcal{B} . Interestingly, a weighted belief base also cannot, by definition (i.e. due to the pairwise inconsistency), contain more formulas than possible worlds. Indeed, since the revision operator \circ_s transforms weighted belief bases into weighted belief bases, even for small $\mathcal{A}t$ we never have more formulas in the syntactic representation than possible worlds.

We now verify requirement (1): the syntactic revision needs to ensure that the set of classical formulas after revision remain pairwise inconsistent, i.e. the result is a weighted belief base.

Lemma 1. *Let \mathcal{B} be a weighted belief base and let (φ, m) be an input. For all $(\psi_i, m_i), (\psi_j, m_j) \in \mathcal{B}$ we have that $\psi_i \wedge \psi_j$ is not satisfiable. This condition also holds for $\mathcal{B}' = \mathcal{B} \circ_s(\varphi, m)$.*

Proof: For an arbitrary propositional formula ρ we trivially have that $\text{Mod}(\rho) \neq \emptyset$ when ρ is satisfiable, $\text{Mod}(\rho) \cap \text{Mod}(\neg \rho) = \emptyset$ (i.e. a formula and its negation do not share models) and $\text{Mod}(\rho) \cup \text{Mod}(\neg \rho) = \Omega$ (i.e. the models of a formula and its negation span all possible worlds).

We first show that modifying the existing weighted formulas in \mathcal{B}_φ , defined as \mathcal{B}' in Definition 6, does not violate the condition imposed by a weighted belief base. For every weighted formula $(\psi, m) \in \mathcal{B}_\varphi$ we have that $\psi \wedge \varphi$ is consistent due to Definition 6, i.e. we have that $\text{Mod}(\psi) \cap \text{Mod}(\varphi) \neq \emptyset$. Since $\text{Mod}(\psi) \subseteq \Omega$ with $\Omega = (\text{Mod}(\varphi) \cup \text{Mod}(\neg \varphi))$ and since $\text{Mod}(\varphi) \cap \text{Mod}(\neg \varphi) = \emptyset$ we have that $\text{Mod}(\psi)$ can be rewritten as $\text{Mod}(\psi \wedge \varphi) \cup \text{Mod}(\psi \wedge \neg \varphi)$ with $\text{Mod}(\psi \wedge \varphi) \cap \text{Mod}(\psi \wedge \neg \varphi) = \emptyset$. We thus have that the conjunction of $(\psi \wedge \varphi)$ and $(\psi \wedge \neg \varphi)$ is not satisfiable as these formulas do not share any models. In addition, for every weighted formula $(\psi', m) \in \mathcal{B} \setminus \mathcal{B}_\varphi$ the condition holds by assumption for \mathcal{B} and since $\text{Mod}(\psi') \cap \text{Mod}(\varphi) = \emptyset$. We thus know that the condition is satisfied for \mathcal{B}' in Definition 6.

It remains to be checked whether $((\varphi \wedge \neg \bigvee(\mathcal{B}_\varphi^*)), m)$ upholds the condition. Since $\text{Mod}(\rho) \cap \text{Mod}(\neg \rho) = \emptyset$ and $\text{Mod}(\rho) \cup \text{Mod}(\neg \rho) = \Omega$ with $\rho = \bigvee(\mathcal{B}_\varphi^*)$ we know that we can rewrite $\text{Mod}(\varphi)$ as $\text{Mod}(\varphi \wedge \bigvee(\mathcal{B}_\varphi^*)) \cup \text{Mod}(\varphi \wedge \neg \bigvee(\mathcal{B}_\varphi^*))$ where all models in $\text{Mod}(\varphi \wedge \bigvee(\mathcal{B}_\varphi^*))$ are considered in the case discussed above and, trivially, $\text{Mod}(\varphi \wedge \neg \bigvee(\mathcal{B}_\varphi^*)) \cap \text{Mod}(\psi) = \emptyset$ for all $(\psi, m) \in \mathcal{B}$ since no models are shared with the negation of a formula. ■

We now focus our attention on verifying requirement (2): the revision result obtained using the syntactic revision opera-

tor must correspond to the result obtained using the semantic revision operator. Since Lemma 1 shows us that none of the weighted formulas in a belief base \mathcal{B} share any models, we can simplify this requirement. Indeed, it suffices to verify that the weight associated with each formula in the syntactic approach corresponds exactly with the weight associated with its models (i.e. possible worlds) in the semantical approach. In other words, we need to verify that $\Phi(\omega) = m$ with $\omega \in \text{Mod}(\psi)$ iff $(\psi, m) \in \mathcal{B}$ and otherwise $\Phi(\omega) = 0$. Similar as before, we will use the notation $\mathcal{B} \circ_s I$ with $I = \langle i_1, \dots, i_n \rangle$ a sequence of inputs to denote $\mathcal{B} \circ_s i_1 \circ_s \dots \circ_s i_n$.

Proposition 3. *Let I be a finite sequence of inputs. Let Φ_0 be the epistemic state such that $\forall \omega \in \Omega$ we have that $\Phi_0(\omega) = 0$. Let $\mathcal{B}_0 = \{\}$ and let $\Phi_n = \Phi_0 \circ I$ and $\mathcal{B}_n = \mathcal{B}_0 \circ_s I$. We have that $(\psi, m) \in \mathcal{B}_n$ iff $\Phi_n(\omega) = m$ for every $\omega \in \text{Mod}(\psi)$.*

Proof: We prove this by induction on the number of inputs in I . The base cases are trivial. When $I = \langle \rangle$ there is nothing to do and the proposition holds vacantly. When I contains only a single element, i.e. $I = \langle (\psi_1, m_1) \rangle$, then we trivially obtain that $\mathcal{B}_1 = \{(\psi_1, m_1)\}$ while $\Phi_1(\omega) = m_1$ iff $\omega \models \psi_1$ and $\Phi_1(\omega) = 0$ otherwise. This is exactly the epistemic state induced by \mathcal{B}_1 according to Definition 5.

Now assume that for a sequence of inputs I of size n with $\mathcal{B}_n = \mathcal{B}_0 \circ_s I$ and $\Phi_n = \Phi_0 \circ I$ we have that $\Phi_n(\omega) = m$ with $\omega \in \text{Mod}(\psi)$ iff $(\psi, m) \in \mathcal{B}_n$. Equivalently, we thus have that $\forall \omega \in \Omega \cdot \Phi_n(\omega) = \Psi_n(\omega)$ with Ψ_n the epistemic state induced by \mathcal{B}_n in Definition 5. We show that this equivalence is upheld after revising with the $(n+1)$ -th input $in = (\varphi_{n+1}, \mu_{n+1})$ where $\Phi_{n+1} = \Phi_n \circ in$ and $\mathcal{B}_{n+1} = \mathcal{B}_n \circ_s in$.

By definition of a simple epistemic state and the semantical revision operator we know that $\Phi_{n+1}(\omega) = \Phi_n(\omega)$ for all ω such that $\omega \not\models \varphi_{n+1}$. Similarly, in \mathcal{B}_{n+1} we have the formulas $\mathcal{B}_n \setminus (\mathcal{B}_n)_{\varphi_{n+1}}$ and the formulas $(\psi \wedge \neg \varphi_{n+1}, m)$ for those $(\psi, m) \in \mathcal{B}_n$ such that $\psi \wedge \varphi_{n+1}$ is consistent. In other words: the weight of formulas that do not have models in common with φ_{n+1} as well as the weight of the models of $\psi \wedge \neg \varphi_{n+1}$ (those models of ψ that are not models of φ_{n+1}) is unchanged.

For all ω such that $\omega \models \varphi_{n+1}$ we know that $\Phi_{n+1}(\omega) = \Phi_n(\omega) + \mu_{n+1}$. Similarly, in \mathcal{B}_{n+1} we have the formulas $(\psi \wedge \varphi_{n+1}, m + \mu_{n+1})$ for those $(\psi, m) \in \mathcal{B}_n$ such that $\psi \wedge \varphi_{n+1}$ is consistent. However, there may be models in $\text{Mod}(\varphi_{n+1})$ that are not yet in \mathcal{B}_n . Indeed, these models are exactly the models of the formula $\psi_{n+1} \wedge \neg \bigvee ((\mathcal{B}_n)_\varphi^*)$ to which we assign the weight μ_{n+1} . Once again, it is easy to verify from Definition 5 that we thus find $\forall \omega \in \Omega \cdot \Phi_{n+1}(\omega) = \Psi_{n+1}(\omega)$ with Ψ_{n+1} the epistemic state induced by \mathcal{B}_{n+1} . ■

The syntactic revision operator introduced in Definition 6 thus behaves as expected. It transforms a weighted belief base into another weighted belief base, for which it is easy to determine the λ -value of any arbitrary formula. Furthermore, we just proved that revision with any arbitrary formula indeed gives the correct results, i.e. it corresponds with the results that we would obtain using the semantical operator.

So far we have not considered whether formulas expressed in the language \mathcal{L}_\geq can be entailed. However, since we know how to determine the weight associated with a formula $\varphi \in \mathcal{L}$, it is straightforward to extend our approach to contexts in the

language \mathcal{L}_\geq . Indeed, we can readily apply $\text{pare}(\varphi \geq \psi)$ from Section II. Furthermore, verifying whether $\varphi \geq \psi$ holds only requires $|\mathcal{B}|+1$ satisfiability checks when \mathcal{B} is sorted according to the weight of its formulas. Once we find the highest weight μ for φ , we only need to verify if $\psi \wedge \psi'$ is satisfiable for $(\psi', \mu') \in \mathcal{B}$ with $\mu' \leq \mu$. Reasoning about the relationship of the plausibility of two formulas $\varphi \in \mathcal{L}$ can thus be done as efficiently as determining the weight of a formula $\varphi \in \mathcal{L}$.

IV. TRACTABLE REVISION

In the previous section we introduced – to the best of our knowledge – the first syntactic operator capable of dealing with arbitrary unreliable inputs. However, in many situations, we may be dealing with resource-bounded agents or agents for which reactivity is of paramount importance. In such cases, the NP-hard syntactic approach from the previous section may not be feasible or may not be sufficiently fast.

In this section, we develop a tractable approach to belief change with unreliable inputs. To attain a tractable approach we impose a number of restrictions. A common restriction in the literature is to only consider literals as inputs (e.g. [2]). Based on this restriction, a simple syntactical representation can be envisaged where we only need to keep track of the weights $\bar{\mu}$ and $\underline{\mu}$ associated with each atom $a \in \text{At}$, with $\bar{\mu}$ and $\underline{\mu}$ the weight of resp. a and $\neg a$. We have:

Definition 7. A compact epistemic state \mathcal{W} is a mapping $\mathcal{W} : \text{At} \rightarrow (\mathbb{Z}, \mathbb{Z})$ with $\mathcal{W}(a) = (\bar{\mu}, \underline{\mu})$, i.e. the weight associated with resp. a and $\neg a$.

The epistemic state $\Phi_{\mathcal{W}}$ associated with a compact epistemic state \mathcal{W} is defined next. We denote the weight of a literal $l \in \text{Lit}$ given a compact epistemic state \mathcal{W} as $w_{\mathcal{W}}(l) = \bar{\mu}$ if $l = a$ and $w_{\mathcal{W}}(l) = \underline{\mu}$ if $l = \neg a$ with $\mathcal{W}(a) = (\bar{\mu}, \underline{\mu})$.

Definition 8. Let \mathcal{W} be a compact epistemic state. We have that $\Phi_{\mathcal{W}}$, defined as

$$\Phi_{\mathcal{W}}(\omega) = \sum_{\omega \models l} w_{\mathcal{W}}(l)$$

is the corresponding (semantical) epistemic state.

This definition is similar to Definition 5 from the previous section, where we now only use a set of literals (which can share models) as a compact representation of an epistemic state. Given the way we defined a compact epistemic state, and since we only allow (weighted) literals as input, a tractable belief change operator only has to update the weight of the literal given in the input. We have:

Definition 9. Let \mathcal{W} be a compact epistemic state. Let (l, μ) be an input with $l \in \text{Lit}$. We define $\mathcal{W}' = \mathcal{W} \circ_t (l, \mu)$ as:

$$\mathcal{W}'(a) = \begin{cases} (\bar{\mu} + \mu, \underline{\mu}) & \text{if } l = a \\ (\bar{\mu}, \underline{\mu} + \mu) & \text{if } l = \neg a \\ \mathcal{W}(a) & \text{otherwise} \end{cases}$$

with $\mathcal{W}(a) = (\bar{\mu}, \underline{\mu})$.

Proposition 4. *Revising a compact epistemic state can be implemented using an algorithm with $\mathcal{O}(\log_2 |\text{At}|)$ complexity.*

Proof: A compact epistemic state can be implemented as a sorted map with each element being a pair (a, V) . The atom ‘ a ’ is used as the key and V is a pair of values. Belief revision can then be implemented as a binary search over the keys requiring at most $\log_2(|\mathcal{A}t|)$ steps and a constant time update of the respective value. ■

Example 6. Consider an agent who has received no prior input, i.e. $\mathcal{W}(a) = (0, 0)$ for every $a \in \mathcal{A}t$. Assume we successively update \mathcal{W} with the inputs $(\neg c, 2)$, $(a, 4)$, $(b, -3)$, $(a, 1)$, $(c, 2)$ and $(\neg a, 4)$. We obtain \mathcal{W}' with:

$$\mathcal{W}'(a) = (5, 4) \quad \mathcal{W}'(b) = (-3, 0) \quad \mathcal{W}'(c) = (2, 2).$$

Now assume that $\mathcal{W}'' = \mathcal{W}' \circ_t (\neg b, 4)$. From Definition 9 we know that we only have to change the ordered pair associated with b , i.e. we have $\mathcal{W}''(b) = (-3, 4)$. The epistemic state $\Phi_{\mathcal{W}''}$ corresponding with \mathcal{W}'' according to Definition 8 is:

$$\begin{aligned} \Phi_{\mathcal{W}''}(\{a, b, c\}) &= 4 & \Phi_{\mathcal{W}''}(\{\neg a, b, c\}) &= 3 \\ \Phi_{\mathcal{W}''}(\{a, b, \neg c\}) &= 4 & \Phi_{\mathcal{W}''}(\{\neg a, b, \neg c\}) &= 3 \\ \Phi_{\mathcal{W}''}(\{a, \neg b, c\}) &= 11 & \Phi_{\mathcal{W}''}(\{\neg a, \neg b, c\}) &= 10 \\ \Phi_{\mathcal{W}''}(\{a, \neg b, \neg c\}) &= 11 & \Phi_{\mathcal{W}''}(\{\neg a, \neg b, \neg c\}) &= 10 \end{aligned}$$

We now prove the correctness of the operator \circ_t introduced in Definition 9. As before, we use $\mathcal{B} \circ_t I$ with $I = \langle i_1, \dots, i_n \rangle$ a sequence of inputs, each of the form (l, μ) with $l \in \mathcal{L}it$ and $\mu \in \mathbb{Z}$, to denote $\mathcal{B} \circ_t i_1 \circ_t \dots \circ_t i_n$.

Proposition 5. *Let I be a finite sequence of inputs. Let Φ_0 be an epistemic state such that $\forall \omega \in \Omega$ we have that $\Phi_0(\omega) = 0$. Let \mathcal{W}_0 be a compact epistemic state with $\mathcal{W}_0(a) = (0, 0)$ for all $a \in \mathcal{A}t$ and let $\Phi_n = \Phi_0 \circ I$ and $\mathcal{W}_n = \mathcal{W}_0 \circ_t I$. We have that $\Phi_n(\omega) = \Phi_{\mathcal{W}_n}(\omega)$ for all $\omega \in \Omega$ with $\Phi_{\mathcal{W}_n}$ as defined in Definition 8.*

Proof: We prove this by induction on the number of inputs in I . When $I = \langle \rangle$ the proposition holds trivially. When $I = \langle (l_1, \mu_1) \rangle$ we have $\Phi_1(\omega) = \mu_1$ iff $\omega \models l_1$ and $\Phi_1(\omega) = 0$ otherwise. This corresponds exactly with $\Phi_{\mathcal{W}_1}$ since $w_{\mathcal{W}_1}(l_1) = \mu_1$ and $w_{\mathcal{W}_1}(l) = 0$ for all other $l \in \mathcal{L}it$. Now assume that $\Phi_{n-1}(\omega) = \Phi_{\mathcal{W}_{n-1}}(\omega)$ for I a sequence of $n-1$ inputs. We show that the equivalence is upheld after revising with the n -th input $in = (l_n, \mu_n)$.

Note that $\Phi_{n-1}(\omega) \neq \Phi_n(\omega)$ only if $\omega \models l$. In particular, we have that $\Phi_n(\omega) = \Phi_{n-1}(\omega) + \mu_n$. Similarly, $\Phi_{\mathcal{W}_{n-1}}(\omega) \neq \Phi_{\mathcal{W}_n}(\omega)$ only if $\omega \models l_n$. We have that $w_{\mathcal{W}_{n-1}} = w_{\mathcal{W}_n} + \mu_n$ from Definition 9 and thus, due to Definition 8, that $\Phi_{\mathcal{W}_{n-1}}(\omega) = \Phi_{\mathcal{W}_n}(\omega) + \mu_n$. Hence $\Phi_n(\omega) = \Phi_{\mathcal{W}_n}(\omega)$ due to the induction hypotheses and since both are modified with the weight μ_n only if $\omega \models l$. ■

So far, we have seen that restricting ourselves to literals as inputs makes it possible to use a simpler representation that allows for a very efficient belief change operator. Of course, what we are eventually interested in is whether a given formula φ is entailed by the agent’s beliefs, i.e. if the agent believes φ . To this end we can easily define the belief set as $Bel(\mathcal{W}) = \bigwedge \{l \mid l \in \mathcal{L}it, str_{\mathcal{W}}(l) > 0\}$ with $str_{\mathcal{W}}(l) = w_{\mathcal{W}}(l) - w_{\mathcal{W}}(\neg l)$ and \mathcal{W} a compact epistemic state. It is then straightforward to verify that, for an arbitrary

compact epistemic state \mathcal{W} , $Bel(\mathcal{W}) = Bel(\Phi_{\mathcal{W}})$ with $\Phi_{\mathcal{W}}$ the epistemic state induced by \mathcal{W} . Furthermore, since $Bel(\mathcal{W})$ is a conjunction of literals, evaluating whether a formula $\varphi \in \mathcal{L}$ is true according to the belief set of the agent can easily be done by verifying whether φ is true after replacing all occurrences of literals from $Bel(\mathcal{W})$ in φ by \top and all others by \perp .

The problem becomes more intricate when we want to determine the λ -value of formulas. To do so efficiently, we need to restrict the language of the formulas. In particular, we will look at a fragment of the language for which it is easy to determine the *bounded literals*, i.e. those literals for which the weight is known. To see how this can make it easier to determine the λ -value, we will reconsider Example 6 where $\mathcal{A}t = \{a, b, c\}$. To determine the λ -value of the formula $(\neg a \wedge c)$ we are forced to use the weight associated with $\neg a$ and c , i.e. the set of bounded literals is $\{\neg a, c\}$ while ‘ b ’ is unbounded. Similarly, the possible worlds that satisfy the formula $(\neg a \wedge c)$ are $\{\neg a, b, c\}$ and $\{\neg a, \neg b, c\}$. The weight of the formula is then the highest weight associated with either of these possible worlds. From Definition 8 we know that $\Phi(\{\neg a, b, c\}) = 4 + 2 + (-3) = 3$ whereas $\Phi(\{\neg a, \neg b, c\}) = 4 + 2 + 4 = 10$. In other words: to determine the weight of the formula we have to use the weight associated with the bounded literals (i.e. $\neg a$ and c) while we can freely take the maximum weight associated with either the positive or negative atom b , since b was unbounded. As long as the fragment of the language we consider makes it easy and unambiguous to determine the set of bounded literals, we can thus determine the λ -value of a formula in a tractable way.

The fragment $\mathcal{L}_t \subseteq \mathcal{L}$ can be defined in BNF as: ²

$$\begin{aligned} disj &::= a \mid \neg a \mid disj_1 \vee disj_2 \\ conj &::= a \mid \neg a \mid conj_1 \wedge conj_2 \\ \varphi &::= a \mid \neg a \mid disj \wedge conj \mid \varphi_1 \vee \varphi_2 \end{aligned}$$

Intuitively, the language \mathcal{L}_t ensures that whenever a conjunction occurs, one of the branches will be composed of only conjunctions while the other branch will only contain disjunctions. Notice that the fragment \mathcal{L}_t is quite expressive as any formula in DNF is part of this language.

We now define the weight associated with a formula in \mathcal{L}_t . To simplify this definition, we introduce the following notations. For $l \in \mathcal{L}it$ we define $\max_{\mathcal{W}}(l) = \max(\overset{+}{\mu}, \bar{\mu})$ with $\mathcal{W}(l^*) = (\overset{+}{\mu}, \bar{\mu})$, i.e. the maximum weight associated with a given atom or its negation. We define $T_{\mathcal{W}} = \sum_{a \in \mathcal{A}t} \max_{\mathcal{W}}(a)$, i.e. the total of all maximum weights associated with each atom (or: the maximum weight when no literals are bounded). The value $T_{\mathcal{W}}$ can also be computed as a byproduct of belief change. Indeed, for $\mathcal{W}' = \mathcal{W} \circ_t (l, \mu)$ it can easily be computed as $T_{\mathcal{W}'} = T_{\mathcal{W}} - \max_{\mathcal{W}}(l) + \max_{\mathcal{W}'}(l)$.

Definition 10. Let \mathcal{W} be a compact epistemic state and $\varphi \in \mathcal{L}_t$. Let L be a set of literals. We recursively define $\lambda_t(\varphi, L)$ as:

$$\begin{aligned} \lambda_t(\varphi_1 \vee \varphi_2, L) &= \max(\lambda_t(\varphi_1, L), \lambda_t(\varphi_2, L)) \\ \lambda_t(d \wedge c, L) &= \lambda_t(d, L \cup lit(c)) \end{aligned}$$

²While the language does not explicitly allow formulas of the form $c \wedge d$ in φ , any such formula can trivially be converted to the form $d \wedge c$.

$$\lambda_t(l, L) = \begin{cases} -\infty & \text{if } \text{inconst}(L \cup \{l\}) \\ \max_{T_{\mathcal{W}}}(l, L) & \text{otherwise} \end{cases}$$

with $\text{inconst}(S)$ true whenever $\exists a \in \text{At} \cdot \{a, \neg a\} \subseteq S$ and $\max_{T_{\mathcal{W}}}(l, L) = T_{\mathcal{W}} - \sum_{l' \in L \cup \{l\}} (w_{\mathcal{W}}(l') - \max_{\mathcal{W}}(l'))$.

The definition reflects the intuition we described earlier in the section. To evaluate a formula, we need to keep track of the bounded literals. A conjunction bounds variables (i.e. it expresses which literals must be true) while a disjunction takes the maximum of the values of both operands without altering the set of bounded literals. A formula that is (reduced to) just a literal l , is evaluated by considering the set $L \cup \{l\}$, i.e. all literals bounded so far including l . When $L \cup \{l\}$ is inconsistent the weight is $-\infty$. Otherwise, the weight is determined by starting from $T_{\mathcal{W}}$, i.e. the weight of an unbounded formula, and removing from it the maximum weight associated with the bounded literals. The correct weight of each bounded literal is then added to obtain the λ -value of the formula.

Example 7. Consider the formula $\varphi = (a \vee \neg c) \wedge (c \wedge b)$ and \mathcal{W}'' from Example 6. We have:

$$\begin{aligned} \lambda_t(\varphi, \emptyset) &= \lambda_t((a \vee \neg c) \wedge (c \wedge b), \emptyset) \\ &= \lambda_t(a \vee \neg c, \{b, c\}) \\ &= \max \{ \lambda_t(a, \{b, c\}), \lambda_t(\neg c, \{b, c\}) \} \\ &= \max \{ \lambda_t(a, \{b, c\}), -\infty \} \\ &\quad (\text{since } \{c, \neg c\} \in (\{b, c\} \cup \{\neg c\})) \\ &= \lambda_t(a, \{b, c\}) \\ &= T_{\mathcal{W}} - (5 - 5) - (4 - (-3)) - (2 - 2) = 4 \end{aligned}$$

We now show that $\lambda_t(\varphi, \emptyset)$ corresponds with $\lambda(\varphi)$, i.e. the weight of a formula on the semantical level:

Proposition 6. *Let \mathcal{W} be a compact epistemic state and $\Phi_{\mathcal{W}}$ the epistemic state induced by \mathcal{W} . We have that $\lambda_t(\varphi, \emptyset) = \lambda(\varphi)$ with λ_t as in Definition 10.*

Proof: We first consider formulas without conjunction. Since the weight of a disjunction is the maximum of its constituents, we only need to verify that $\lambda_t(l, \emptyset) = \lambda(l)$. By definition, $T_{\mathcal{W}}$ is the highest weight associated with any (set of) possible world(s). Either l is entailed by a world with the highest weight, in which case $w_{\mathcal{W}}(l) \geq w_{\mathcal{W}}(\neg l)$, i.e. $T_{\mathcal{W}} - \max_{\mathcal{W}}(l) + w_{\mathcal{W}}(l) = T_{\mathcal{W}}$. Otherwise, $w_{\mathcal{W}}(l) < w_{\mathcal{W}}(\neg l)$ and the possible world ω such that $\omega \models l$ is the one that entails all other literals with highest associated weight. As such, the weight of ω is

$$\begin{aligned} &\left(\sum_{a \in \text{At} \setminus \{l^*\}} \max_{\mathcal{W}}(a) \right) + w_{\mathcal{W}}(l) \\ &\text{with } \sum_{a \in \text{At} \setminus \{l^*\}} \max_{\mathcal{W}}(a) = T_{\mathcal{W}} - \max_{\mathcal{W}}(l) \end{aligned}$$

by definition of $T_{\mathcal{W}}$ and Definition 8.

We now consider the evaluation of a formula of the form $c \wedge d$ or, equivalently, $(\varphi_1 \wedge \dots \wedge \varphi_n) \wedge (\psi_1 \vee \dots \vee \psi_m)$. Using the distributive law, we can rewrite this as $(\varphi_1 \wedge \dots \wedge \varphi_n \wedge \psi_1) \vee \dots \vee (\varphi_1 \wedge \dots \wedge \varphi_n \wedge \psi_m)$. We thus need to verify whether the

weight of a formula of the form $\theta = (\varphi_1 \wedge \dots \wedge \varphi_n \wedge \psi_i)$ is correctly determined. The possible world ω such that $\omega \models \theta$ with the highest associated weight is the one that entails all literals not found in θ with the maximum associated weight. We have that the weight of ω is given by

$$\begin{aligned} &\sum_{a \in \text{At} \setminus \text{lit}^*(\theta)} \max_{\mathcal{W}}(a) + \sum_{l \in \text{lit}(\theta)} w_{\mathcal{W}}(l) \\ &\text{with } \sum_{a \in \text{At} \setminus \text{lit}^*(\theta)} \max_{\mathcal{W}}(a) = T_{\mathcal{W}} - \sum_{l \in \text{lit}(\theta)} \max_{\mathcal{W}}(l) \end{aligned}$$

where $\text{lit}^*(\theta) = \{l^* \mid l \in \text{lit}(\theta)\}$. Finally, we consider the situation where $\text{lit}(\theta)$ is inconsistent, i.e. there exists an $a \in \text{At}$ such that $\{a, \neg a\} \subseteq \text{lit}(\theta)$. We then have $\lambda(\theta) = -\infty$ and, correspondingly, Definition 10 returns $-\infty$. ■

Proposition 7. *Computing the λ -value of a formula $\varphi \in \mathcal{L}_t$ using a compact epistemic state can be implemented using an algorithm with $\mathcal{O}(k \cdot \log_2 |\text{At}|)$ complexity where k is the number of literals in φ .*

Proof: An algorithm can straightforwardly be devised based on Definition 10 that traverses a given formula tree and collects the bounded literals in each conjunctive branch. Such a traversal is linear in the size of the formula. Once the set of bounded literals has been determined, the λ -value can be computed by retrieving the n distinct literals found in the bounded branch. Assuming that the compact epistemic state is encoded as a sorted map, retrieving the value of each literal is $\mathcal{O}(\log_2 |\text{At}|)$. In the worst case, the value of all literals in the formula need to be determined. We thus need to retrieve at most k values where k is the number of literals in φ . ■

Similar as in the previous section, it is possible to extend the language \mathcal{L}_t to a language \mathcal{L}_t^{\geq} for which it is easy to evaluate formulas of the form $\varphi \geq \psi$ or $\varphi > \psi$. We have seen that evaluating a formula $\varphi \in \mathcal{L}_t$ is tractable. Once we know the value $\lambda(\varphi)$ of a formula φ , we can readily apply $\text{pare}(\varphi \geq \psi)$ from Section II. As such, we define \mathcal{L}_t^{\geq} in BNF as ³:

$$\begin{aligned} \text{disj} &::= a \mid \neg a \mid \text{disj}_1 \vee \text{disj}_2 \\ \text{conj} &::= a \mid \neg a \mid \text{conj}_1 \wedge \text{conj}_2 \\ \varphi &::= a \mid \neg a \mid \text{disj} \wedge \text{conj} \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \geq \varphi_2 \mid \varphi_1 > \varphi_2 \end{aligned}$$

Example 8. Consider \mathcal{W}'' from Example 6. We have that $c \geq a \wedge b$ since $\lambda(c) = 11$ and $\lambda(a \wedge b) = 11 - 5 - 4 + 5 + (-3) = 4$. Similarly, we can verify that $(c > b) \vee (c > a)$ since $11 > 4$ and $11 \not> 11$, i.e. we have $\top \vee \perp$ or $\max\{\top, \perp\} = \top$.

V. RELATED WORK

Syntactic operators for revision with either classical or uncertain inputs have been considered in the literature, where most deal with classical inputs and are based on the AGM style of revision (e.g. [1], [15]). Syntactic operators that are able to deal with iterated belief revision, which is most often defined on a semantical level, are far less common. Operators for iterated belief change are usually defined on the level of

³See <http://www.eecs.qub.ac.uk/~k.bauters/ictai14> for a Python implementation of the tractable syntactic approach presented in this paper.

epistemic states, for which we can use Ordinal Conditional Functions (OCF) [17] or, for example, the representation we used in this paper based on [14]. A syntactic representation for OCF [17], along with the conditions that such a representation has to satisfy, was presented in [19].

A syntactic revision operator in the setting of possibility theory was later presented in [4] that could also deal with uncertain input. This operator makes use of the ability to transform a possibility distribution into an OCF and vice versa, effectively developing a revision operator for both frameworks based on the earlier work. However, this approach treats uncertain input as a form of conditioning, where the resulting beliefs have to conclude the formula with exactly the given degree of uncertainty. They do not interpret uncertain inputs as in this paper, i.e. as information that strengthens or weakens the beliefs that the agent currently holds. Postulates for how to reasonably treat uncertain input as unreliable information were only presented later in the literature [14].

Interesting work in the BDI setting has been done in [2], where the authors develop a tractable form of belief revision by devising a cross-over between AGM style revision and reason-maintenance style belief revision [9]. In particular, in a BDI setting where beliefs are modelled by literals (and plans take their usual form of rules) it can be shown that the operator satisfies most of the AGM postulates. Still, this approach can only deal with classical input. A framework for BDI agents dealing with uncertain input has been presented in [5], where the authors develop a theoretical framework based on possibility theory where both beliefs and desires are represented as possibility distributions. Confusingly, [14] used the term belief change that was also used in [5], but both frameworks are distinct. In [5] they then develop a way to select the best set of goals to be adapted depending on the consistency of these goals, which in turn depends on the uncertain beliefs of the agent. Their work was extended in [6] where they developed a syntactic approach for their framework, highlighting the practical feasibility. However, as discussed, their work is based on the notion of interpreting uncertain input in the sense of conditioning, where the beliefs of the agent need to exactly reflect the input uncertainty.

VI. CONCLUSIONS

To the best of our knowledge this paper is the first to introduce a syntactic approach to belief change as defined in [14] for dealing with unreliable input. Our approach can easily be implemented using off-the-shelf satisfiability solvers. We proved the correctness of the approach and showed how we can extend it to a more expressive language that allows an agent to reason over the plausibility of its beliefs. With the advent of satisfiability solvers that work directly on NNF formulas [12] our approach furthermore promises to be fast. An experimental evaluation of the approach is, however, left for future work. For situations where we are dealing with a resource-bounded agent, we also proposed a tractable approach to belief change. To obtain a tractable approach, some restrictions were imposed. Particularly, inputs are limited to weighted literals (a restriction common in other tractable

approaches) and determining the weight of arbitrary formulas is only feasible for fragments of the propositional language. Still, the fragment we identified allows for formulas in DNF to be efficiently evaluated. Belief change in the tractable approach can be implemented using a $O(\log_2 |\mathcal{At}|)$ algorithm with $|\mathcal{At}|$ the number of atoms under consideration, while the evaluation of a formula in the fragment we identified can be implemented in a $O(k \cdot \log_2 |\mathcal{At}|)$ algorithm with k the number of unique literals in the formula to be evaluated.

REFERENCES

- [1] C. E. Alchourrón, P. Gärdenfors, and D. Makinson, “On the logic of theory change: Partial meet contraction and revision functions,” *Journal of Symbolic Logic*, vol. 50, no. 2, pp. 510–530, 1985.
- [2] N. Alechina, M. Jago, and B. Logan, “Resource-bounded belief revision and contraction,” in *Proc. of DALI’05*. Springer-Verlag, 2005, pp. 141–154.
- [3] K. Bauters, W. Liu, J. Hong, C. Sierra, and L. Godo, “Can(plan)+: Extending the operational semantics of the BDI architecture to deal with uncertain information,” 2014, accepted for UAI’14.
- [4] S. Benferhat, D. Dubois, H. Prade, and M.-A. Williams, “A practical approach to revising prioritized knowledge bases,” *Studia Logica*, vol. 70, no. 1, pp. 105–130, 2002.
- [5] C. da Costa Pereira and A. Tettamanzi, “An integrated possibilistic framework for goal generation in cognitive agents,” in *Proc. of AAMAS’10*, 2010, pp. 1239–1246.
- [6] —, “A syntactic possibilistic belief change operator: Theory and empirical study,” *Web Intelligence and Agent Systems*, vol. 12, no. 2, pp. 155–169, 2014.
- [7] A. Darwiche and M. Goldszmidt, “On the relation between kappa calculus and probabilistic reasoning,” in *Proc. of UAI’94*, 1994, pp. 145–153.
- [8] A. Darwiche and J. Pearl, “On the logic of iterated belief revision,” *Artificial Intelligence*, vol. 89, no. 1-2, pp. 1–29, 1997.
- [9] J. Doyle, “A truth maintenance system,” *Artificial Intelligence*, vol. 12, no. 3, pp. 231 – 272, 1979.
- [10] D. Dubois, S. Moral, and H. Prade, “Belief change rules in ordinal and numerical uncertainty theories,” in *Belief Change*, ser. Handbook of Defeasible Reasoning and Uncertainty Management Systems, D. Dubois and H. Prade, Eds. Springer Netherlands, 1998, vol. 3, pp. 311–392.
- [11] D. Dubois and H. Prade, “Possibility theory as a basis for qualitative decision theory,” in *Proc. of IJCAI’95*, 1995.
- [12] H. Jain and E. M. Clarke, “Efficient SAT solving for non-clausal formulas using DPLL, graphs, and watched cuts,” in *Proceedings of DAC’09*, 2009, pp. 563–568.
- [13] N. Laverny and J. Lang, “From knowledge-based programs to graded belief-based programs, Part I: On-line reasoning*,” *Synthese*, vol. 147, no. 2, pp. 277–321, 2005.
- [14] J. Ma and W. Liu, “A framework for managing uncertain inputs: An axiomization of rewarding,” *International Journal of Approximate Reasoning*, vol. 52, no. 7, pp. 917–934, 2011.
- [15] B. Nebel, “Base revision operations and schemes: Semantics, representation and complexity,” in *Proc. of ECAI’94*, 1994, pp. 341–345.
- [16] A. Rao and M. Georgeff, “Modeling rational agents within a BDI-architecture,” in *Proc. of KR’91*, 1991, pp. 473–484.
- [17] W. Spohn, “Ordinal conditional functions: A dynamic theory of epistemic states,” in *Causation, Coherence, and Concepts: Proceedings of the Irvine Conference on Probability and Causation*, 1988, pp. 105–134.
- [18] M.-A. Williams, “Transmutations of knowledge systems,” in *Proc. of KR’94*, 1994, pp. 619–629.
- [19] —, “Iterated theory base change: A computational model,” in *Proc. of IJCAI’95*, 1995, pp. 1541–1550.