User insisted redistribution of belief in hierarchical classification spaces

Wilbert van Norden
Defence Materiel Organisation
CAMS – Force Vision
Den Helder, the Netherlands
w.l.van.norden@forcevision.nl

Catholijn Jonker
Man-Machine Interaction Group
Delft University of Technology
Delft, the Netherlands
c.m.jonker@tudelft.nl

***Abstract*—Where users and agents, each with their own world model and expertise, work together it is essential to interpret both their beliefs correctly. It is therefore important to keep track of the differences of opinion that occur in such a way that it is understandable for both the agents as well as the user. This paper proposes a generic and flexible way for the user to interact with agents using a integrated world model. To enforce the user's opinion a User Preference Redistribution rule (UPR) is proposed. Through a realistic numerical example we show the validity of this model and the new UPR in contrast to other belief conditioning rules.**

***Index Terms*—Conflict redistribution, Belief Conditioning, Human-Agent Interaction**

I. Introduction

In many situated problems, information from different sources needs to be combined in order to find a solution. A general example of this is a system in which an agent must interact with another agent (as in a Multi-Agent System (MAS)) or with an user. This interaction could be in the form of cooperation; i.e., they work together to find a solution. It could also be that the MAS tries to find a solution and that the user only indicates specific regions of interest within the solution space. This paper focusses on the latter situation where the user may exclude parts in a hierarchical solution space. In Fig. 1 the architecture is represented where the user interacts with a dedicated agent that combines the agents' belief with the user's confirmations.

These principles are applied to a classification task where multiple agents assign belief to labels with varying specificity in the solution hierarchy, which is referred to as multi-label learning in [1], [2], and [3]. For the use of MASs in classification see e.g., [4] and [5]. In these different domains of (multi-label) classification, the structure of the solution space is mostly assumed to be fixed. Other approaches like ID3 by Quinlan, [6], and the approach of Taylor et al., [7], use machine learning techniques to model the solution space. Regardless of how the model was obtained, it is usually assumed that once determined, this model is fixed during run-time. In the case of multi-label learning, it is also enforced that each label has exactly one parent label at the next level in the hierarchy.

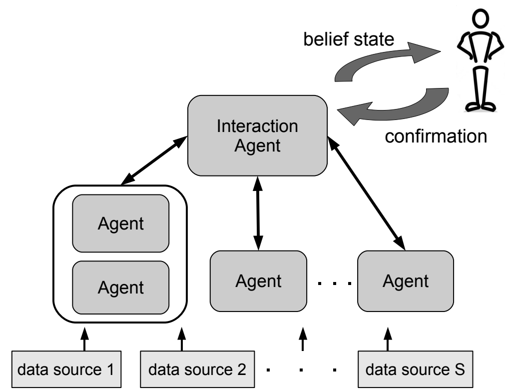


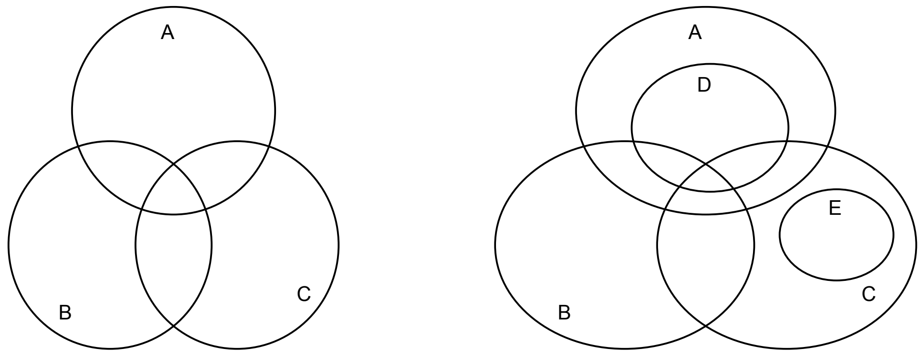
Figure 1. Different agents reason on the incoming data while an interaction agent deals with enforcing user confirmation in the integrated belief state

For the classification of the same objects in a different context, this restriction on the solution space is undesirable. Different users might be interested in different subsets of labels. A flexible model for any hierarchical solution space is required that may be altered at run-time. In order to achieve this flexibility we propose a generic model with multiple hierarchical levels where each element may have no or multiple parents. By using the different subsets for belief conditioning, the desired interaction between the MAS and the user is achieved.

The generic model for a hierarchical solution space is introduced in Section II. Different existing belief conditioning rules are discussed in Section III. Section IV introduces a new conditioning rule for enforcing user confirmations on the solution space based on the generic model. This new approach is tested in Section V for two examples and the results are compared to existing conditioning rules. Section VI closes with the conclusions.

II. The hierarchical model

Fig. 2(a) shows a general solution space for a fully free problem, where elements A, B, and C do not have a hierarchical relationship. Based on this model, we say that all three elements have an equal level of specificity. In hierarchical



(a) A generic free fusion model with *3* elements

(b) Fusion model with fully overlapping elements

Figure 2. Two different fusion models

classification problems, some elements are fully enclosed by others, see e.g., Fig. 2(b). Some elements in Fig. 2(b) are fully overlapping and they therefore do not have the same level of specificity.

The model we propose to use, is based on elements (or labels) with different levels of specificity that may fully overlap. This notion is in line with previous work on classification as discussed in [8]. Here, the domain specific approach from [8] is generalised for any hierarchical solution space. For K specificity levels and N_k elements on specificity level k, with $k \in \{1, 2, \ldots, K\}$, we define the model $\mathcal{M}$ as,

$$\mathcal{M} \ : \ \begin{Bmatrix} \{\theta_{1,1} & \theta_{1,2} & \cdots & \theta_{1,N_1}\} \\ \{\theta_{2,1} & \theta_{2,2} & \cdots & \theta_{2,N_2}\} \\ & & \vdots & \\ \{\theta_{K,1} & \theta_{K,2} & \cdots & \theta_{K,N_K}\} \end{Bmatrix}.$$

Based on this model, the hyper-power set based on Dedekind lattices (see [9]), denoted D^Θ, is constructed using the $\cap$ and $\cup$ operators on all elements from Θ as discussed in [10] and [11]. Since it could be that not all combinations in D^Θ are physically possible, model constraints are contained in set $\varnothing_\mathcal{M}$, where $\varnothing_\mathcal{M} \subseteq \left(D^\Theta \backslash \Theta\right)$ holds. The theoretical case where $\varnothing_\mathcal{M} = \varnothing$ is called the free model since no elements from D^Θ are constrained.

For $\mathcal{M}$ we define all elements on the same specificity level to be mutually exclusive. Though this might seem odd from a technical point of view, it is enforced to more accurately model the way operators view the frame of discernment in e.g., classification. On the same level of specificity an object may be classified as either a *helicopter* or an *airplane* it cannot be both. At a different level however it may be classified as an *air* object which overlaps both of these. On yet another specificity level, the solution might be *fixed wing* or *rotary wing*. In itself these labels are mutually exclusive but each agent may use labels at different specificity labels. The integrated fusion model should incorporate all these overlapping class-labels simultaneously using the same principles.

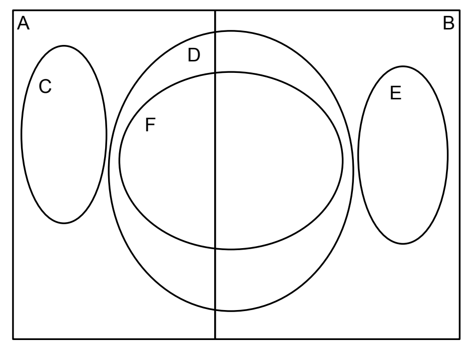


Figure 3. Venn diagram of example frame of discernment with *6* elements

Most intersections between labels are fully overlapping areas between elements on different specificity levels. These overlapping elements in the branch are called child and parent elements. Since they may occur at different specificity levels, the a-th order ancestor of $\theta_{k,n}$ is defined as (1) with $n \in \{1, 2, \ldots, N_k\}$, $v \in \{1, 2, \ldots, N_{k-a}\}$, for $0 \leq k-a \leq K$ and as $\varnothing$ otherwise.

$$\theta_{k,n}^{\uparrow a} = \bowtie \{\theta_{k-a,v} \mid \theta_{k,n} \cap \theta_{k-a,v} \neq \varnothing\} \tag{1}$$

For all $a > 0$ parent elements are found, for $a < 0$ child elements are obtained, and for $a = 0$ the element itself is found. These definitions assume that the rows in the model are ordered based on specificity and $\bowtie$ denotes the join-operator on labels: if $B = \{\beta_1, \ldots, \beta_b\}$ and $D = \{\delta_1, \ldots, \delta_d\}$, then $B \bowtie D = \{\beta_1, \ldots, \beta_b, \delta_1, \ldots, \delta_d\}$.

The obtained model is a hierarchical one and each object may be assigned more than one of these elements. In literature, such types of classification tasks are referred to as multi-label learning, see e.g., [2] and [12]. Here, the modelling of the solution space is similar but differences exist since in our model each element may have more parents at the same specificity level.

Not all ancestors are based on a full overlap. These elements are essential for the conditioning of belief based on the operator preference since they provide a bridge between two labels at the same level of specificity, see e.g., Fig. 3 where element D and its children provide a bridge between elements A and B. Due to this bridging function, these elements are referred to as *bridge* elements.

For notational ease, the variable X refers to labels from Θ or to combinations of labels from D^{Θ}.

In Fig. 3, the Venn diagram of the solution space can be written as model,

$$\mathcal{M} \ : \ \left\{ \begin{matrix} \{A & B\} & \\ \{C & D & E\} \\ \{F\} & & \end{matrix} \right\}.$$

In this model both D and F are bridge elements providing a bridge between their parents A and B.

No classification trees (as in e.g., [13]) have to be constructed using this model for classification and thus there is no chance of getting stuck on a high level node. Another approach to constructing a solution space was discussed by Quinlan, [6], with the ID3 approach and by Taylor et al. These approaches however require machine learning to train the classification solution space. Instead, our model can be used in combination with existing databases of class types as a single level of specification and higher and lower levels may be inserted throughout the process. This provides a more flexible solution space that cannot get stuck on a high level node and where machine learning is not required initially.

III. Existing belief conditioning

The previous section described the solution space with different specificity levels. When multiple agents cooperate, an interaction agent needs to combine their belief based on a fusion algorithm like e.g., Dempster-Shafer theory ([14], [15]), Dezert-Smarandache theory (DSmT, [10]), or any other fusion scheme. In this paper we use DSmT, as was also done in [8], since DSmT requires less adjustments than Dempster-Shafer for hierarchical solution spaces.

A. Model constraints

Combining information using general combination rules causes combined belief to be assigned to combinations of labels from $\varnothing_{\mathcal{M}}$. Since this is undesired, belief assigned to those elements should be redistributed. A known algorithm from DSmT for this redistribution is the Proportional Redistribution Rule number *6* (PCR6) for multiple sources as discussed by Martin and Oswald in [16]. In this article, we use this rule, and the implementation according to algorithm *3* from [16].

For the model constraints $\Theta \backslash \varnothing_{\mathcal{M}} = \Theta$ should hold, i.e., no elements from Θ are contained in $\varnothing_{\mathcal{M}}$, only elements from $D^{\Theta} \backslash \{\Theta \bowtie \varnothing\}$ are possible entries for $\varnothing_{\mathcal{M}}$ as also stated in Section II. Otherwise, PCR6 will give unexpected results: it will assign belief to elements in $\varnothing_{\mathcal{M}}$ since it was not designed to handle such entries in $\varnothing_{\mathcal{M}}$.

Throughout this work $m_c^f(X)$ denotes the amount of belief that was assigned to label X after the combination under the free model and $m_\ell(X)$ denotes the amount of belief that is assigned by agent ℓ to label X. When the combination according to PCR6 is done taking $\varnothing_{\mathcal{M}}$ into account, we denote the combined belief on label X as $m_c^{\text{pcr6}}(X)$ or $m_c(X)$ for short.

When operators indicate some part of Θ to be valid or invalid for an object (positive and negative indications) PCR6 is no longer applicable. A new approach is needed to deal with constraints put on elements from the frame of discernment. For positive indications the Belief Conditioning Rules (BCR) from [17] can be used. However, for negative indications BCRs are not applicable either. For negative indications no redistribution rules could found in the literature.

These operator (or user) constraints contain elements from Θ and are denoted $\varnothing_{\mathcal{U}}$. The interaction agent deals with these constraints separately i.e., in the belief combination rule $\varnothing_{\mathcal{M}}$ is taken into account by PCR6 and later $\varnothing_{\mathcal{U}}$ is enforced. This is done to keep track of where conflict during information fusion is introduced. In other words, during the fusion process we keep track of whether the conflict is caused by one of the information sources or if it is introduced by operator constraints.

Both Shafer's conditioning rules and BCRs can be applied for positive indications by an operator. Giving a negative indication could be seen as a positive indication for the rest of the frame of discernment. We therefore discuss both these methods and compare the new UPR with these methods for negative indications.

B. Shafer's conditioning rule

In [15], a conditioning scheme is proposed to deal with additional information. When a new source indicates that the true solution lies in $X_{\text{true}} \in \Theta$, new believability and plausibility values are determined $\forall X \in \Theta$ according to equations (2) and (3) respectively where $\overline{X_{\text{true}}}$ denotes not X_{true}. Although this approach is simple to implement and to understand, it is not considered to be objective enough as discussed in [17]. Furthermore, with Shafer's conditioning rule we can redistribute plausibility and believability values whereas we also want to redistribute belief masses themselves, making Shafer's rule unfit for our application.

$$\text{Bel}_{\text{new}}(X|X_{\text{true}}) = \frac{\text{Bel}(X \cup \overline{X_{\text{true}}}) - \text{Bel}(\overline{X_{\text{true}}})}{1 - \text{Bel}(\overline{X_{\text{true}}})} \tag{2}$$

$$\text{Pl}_{\text{new}}(X|X_{\text{true}}) = \frac{\text{Pl}(X \cap X_{\text{true}})}{\text{Pl}(X_{\text{true}})} \tag{3}$$

C. Belief conditioning rules

In DSmT, additional information on where the truth is, can be dealt with using the belief conditioning rules (BCR) as explained in [17]. There are numerous variations but all of these are based on three subsets of D^Θ that are constructed using three rules. These three subsets are denoted D_1, D_2, and D_3.

The subset D_1 contains the combination of all labels that are used in the description of where the truth lies. To denote this set of labels, [17] defines the function $s(X_{\text{true}})$ when the truth lies in X_{true}. E.g., when the truth lies in element $X_{\text{true}} = X_1 \cap X_2 \cup X_5$, then $s(X_{\text{true}}) = \{X_1, X_2, X_5\}$. In [17] all combinations of the involved labels with the $\cap$ and $\cup$ operators are denoted as $D_1 = \mathcal{P}_{\mathcal{D}}(X_{\text{true}})$.

The second subset, D_2, is the sub-hyper-power set generated with all labels from $\Theta \backslash s(X_{\text{true}})$ and the $\cap$ and $\cup$ operators when the truth lies in X_{true}.

Finally, the third subset contains all elements from $D^\Theta \backslash \varnothing$ that are not represented in D_1 and D_2. This set is defined as $D_3 = (D^\Theta) \backslash (D_1 \cup D_2)$. All three subsets have no element in common two by two and their union is $D^\Theta \backslash \varnothing$.

All BCR's are based on the redistribution of the masses in D_2 and D_3 to elements in D_1. For BCR1 this is done by proportionally redistributing the combined mass from D_2 and D_3 to the elements in D_1. For the other rules, BCR2–31, redistribution is done directly to particular elements in D_1 or it is done from disjoint subsets of D_2 or D_3 to D_1 and variations thereof, for details see [17].

Consider e.g., the free model of Fig. 2(a) and let the truth be in $A \cup B$. For the different disjoint subsets we then find:

- $D_1 = \{A, B, A \cup B, A \cap B, \}$ and all combinations contained in these elements like e.g., $A \cap B \cap C$;
- $D_2 = \{C\}$ since $s(A \cup B) = \{A, B\}$ and therefore $\Theta \backslash s(A \cup B) = C$;
- $D_3 = \{A \cup C, B \cup C, A \cup B \cup C, C \cup (A \cap B)\}$.

Once these sets are constructed a BCR of choice can be used to redistribute the masses accordingly.

However, assume that in Fig. 2(b) the operator enforces that the solution should be found in $\overline{E}$. In this case $D_3 = \varnothing$ and all elements from $D^\Theta \backslash \varnothing_{\mathcal{M}}$ except E itself are contained in D_1. Thus, the mass from $D_2 = E$ might be redistributed to all or some elements in D_1 depending on the chosen BCR.

A problem with the BCRs arises when e.g., the truth is considered to be in $\overline{D}$ in Fig. 3. The definitions are then no longer clear about what to do with elements like C and E since they could be argued to be a part of D_1 (they could be part of the description of where the truth lies, $C \cup (A \cap \overline{D}) \cup (B \cap \overline{D}) \cup E$, but also to be part of D_2 when the truth is described as $(A \cap \overline{D}) \cup (B \cap \overline{D})$. This ambiguity is caused by the fact that BCR is developed for free models like shown in Fig. 2(a).

IV. User confirmation

In systems where an automated system and an operator work together to find a solution, that operator requires different means of exerting his influence. The operator might e.g., indicate where belief should be held (positive confirmation) or where it should not be held (negative confirmation). Theses types of constraints are not dealt with in PCR6 since that is designed for constraints on elements from $D^\Theta \backslash \Theta$ whereas $\varnothing_{\mathcal{U}}$ contains elements from Θ itself. Shafer's conditioning rule should be able to deal with this but this rule is not considered to be objective enough, as explained in [17], nor does it operate on belief masses.

A. Bridging elements

In [17], several BCR's are proposed to deal with additional information, whether this comes from an operator or another source. However, these BCR rules deal with the situation where a source indicates where belief should be held. Here, we propose a different conditioning rule that deals with the situation where elements from the frame of discernment are excluded. The excluded elements are contained in $\varnothing_{\mathcal{U}}$. The conflict $\mathcal{K}$ that an element $X \in \varnothing_{\mathcal{U}}$ introduces, is the amount of belief it was originally assigned to by the conditioning rule that was used: $\mathcal{K}(X) = m_c(X)$.

Since, in our case the operator says that these elements should be excluded from the model, the new User Preference Redistribution rule (UPR) states that all these constrained elements, and of course their child elements, are assigned zero belief, i.e., $m_c^{\text{upr}}(X) = 0$, $\forall X \in \varnothing_{\mathcal{U}}$. In order to maintain validity in the operator constraints, $\forall X \in \varnothing_{\mathcal{U}}$ there is no $X_i \in \Theta \backslash \varnothing_{\mathcal{U}}$ for which $X_i \cap X = X_i$ holds. This means that if we would constrain D in Fig. 3, then $\varnothing_{\mathcal{U}} = \{D, F\}$ since $D \cap F = F$. The total amount of conflict introduced by $\varnothing_{\mathcal{U}}$ is found by $\sum_{X_j \in \varnothing_{\mathcal{U}}} \mathcal{K}(X_j)$.

$$\sum_{\forall X_\ell \in D^\Theta} m_c^{\text{upr}}(X_\ell) = 1 \tag{4}$$

Since (4) should still hold, the masses that are discarded need to be redistributed. The question becomes, where should it be redistributed to?

The first choice is to redistribute the conflicting mass to one of the parent labels of the excluded one with smallest DSm cardinality that is not excluded itself. In order to find this label we first define $X^\uparrow$ as the set containing all parent labels of $X \in \Theta$. This set joins all $X_i \in \Theta$ for which $X \cap X_i = X$ holds, see (5) with $X \neq X_i$. E.g., in Fig. 3 we say that $C^\uparrow = A$. The smallest parent may be found using the minimum DSm cardinal since the model enforces that a parent always has a larger DSm cardinality than its child element. Note that this definition of a parent differs from (1) since here, only fully overlapping parents are considered.

In [11] the DSm cardinality is explained in detail but in short, this cardinality expresses the number of disjoint parts in the Venn diagram that together form X and it is denoted $\mathcal{C}_{\mathcal{M}}(X)$.

In Fig. 3 e.g., this means that since

$A = (\ (A \cap \overline{C} \cap \overline{D})\ \cup\ (C)\ \cup\ ((D \cap A) \cap \overline{F})\ \cup\ (F \cap A)\ ),$

the DSm cardinal of A is $\mathcal{C}_{\mathcal{M}}(A) = 4$ since it has four disjoint parts in the Venn diagram. Likewise, $\mathcal{C}_{\mathcal{M}}(D) = 4$ and $\mathcal{C}_{\mathcal{M}}(F) = 2$ holds for Fig. 3.

$$X^{\uparrow} = \bowtie \{X_i \in \Theta \mid X_i \cap X = X\ \wedge\ X_i \neq X\} \quad (5)$$

$$X^{\sqcap} = \bowtie \{X_i \in \Theta \mid X_i \cap X \neq \varnothing\ \wedge\ X_i \cap X \neq X\} \quad (6)$$

The set of bridging labels of X, denoted $X^{\sqcap}$, is determined by joining the labels that have a non-empty intersection but that are not fully enclosed by X, (6). E.g., in Fig. 3 $A^{\sqcap} = \{D, F\}$. Combined with $X^{\uparrow}$ all ancestor elements for $a > 0$ are found. This distinction between full parent labels and bridges is made since full parents are the first choice to redistribute belief to, the bridges are the second choice.

It could be that all possible bridges are constrained by the user. In this case we use the constrained bridges to find an unconstrained labels. Say e.g., that $\varnothing_{\mathcal{U}} = \{A, D\}$ in the example of Fig. 3. For element A we find $A^{\uparrow} = \varnothing$ and $A^{\sqcap} \backslash \varnothing_{\mathcal{U}} = \varnothing$ (since $F \cap D = F$ and $D \in \varnothing_{\mathcal{U}}$) and for element D we find $D^{\uparrow} = \varnothing$ and $D^{\sqcap} = B$. We then say that since $A \cap D \neq \varnothing$ and since $B \cap D \neq \varnothing$ that $A^{\rightharpoonup} = B$. In general, we define $X^{\rightharpoonup}$ by (7) with $X_i \in \Theta$.

$$X^{\rightharpoonup} = \bowtie \{X_i \in \Theta \mid X_i \cap X^{\sqcap} \neq \varnothing\ \wedge\ X_i \cap X = \varnothing\} \quad (7)$$

Should it occur that even this $X^{\rightharpoonup} \backslash \varnothing_{\mathcal{U}} = \varnothing$, we then redistribute $\mathcal{K}(X)$ to $\Theta \backslash \varnothing_{\mathcal{U}}$.

We now have four possible areas where the mass could be redistributed to. The area that will be used for the redistribution of the belief assigned to the now constrained element X is denoted X^* and is determined by (8). In essence, the first choice is to use the smallest full parent. If all full parents receive negative confirmation from the user, bridges are chosen. Should these also receive negative confirmation, the bridges are used to find generic labels that have no intersection with the constrained label. Finally, should these labels also receive negative confirmation, the mass is redistributed to all labels from Θ that are not in $\varnothing_{\mathcal{U}}$.

$$X^* = \begin{cases} \min\limits_{X_i \in (X^{\uparrow} \backslash \varnothing_{\mathcal{U}})} \mathcal{C}_{\mathcal{M}}(X_i) & \text{if } X^{\uparrow} \backslash \varnothing_{\mathcal{U}} \neq \varnothing \\ \bigcup\limits_{X_i \in (X^{\sqcap} \backslash \varnothing_{\mathcal{U}})} X_i & \text{if } \begin{array}{l} X^{\uparrow} \backslash \varnothing_{\mathcal{U}} = \varnothing \\ X^{\sqcap} \backslash \varnothing_{\mathcal{U}} \neq \varnothing \end{array} \\ \max\limits_{X_i \in (X^{\rightharpoonup} \backslash \varnothing_{\mathcal{U}})} \mathcal{C}_{\mathcal{M}}(X_i) & \text{if } \begin{array}{l} X^{\uparrow} \backslash \varnothing_{\mathcal{U}} = \varnothing \\ X^{\sqcap} \backslash \varnothing_{\mathcal{U}} = \varnothing \\ X^{\rightharpoonup} \backslash \varnothing_{\mathcal{U}} \neq \varnothing \end{array} \\ \Theta \backslash \varnothing_{\mathcal{U}} & \text{otherwise} \end{cases} \quad (8)$$

Using (8) we can determine where conflicting mass is to be redistributed. Of importance for the runtime behaviour of the algorithm is that the most computationally intensive part — i.e., calculating $X^{\uparrow}$, $X^{\sqcap}$, and $X^{\rightharpoonup}$ — can be done off-line $\forall X \in \Theta$. In contrast, for BCR the disjoint sets are based on $\varnothing_{\mathcal{U}}$ which is presented by the user at runtime, enforcing the computation of BCR to be done during runtime. When UPR is used in an on-line system and $\varnothing_{\mathcal{U}} \neq \varnothing$, (8) can be run with low computational costs.

Having identified where conflict should be redistributed to, the next step is to determine how to redistribute.

B. Redistribution of conflict

Since our approach is redistribution, each element that is unconstrained keeps the belief value that was assigned to them by the combination rule (in our case by PCR6). Belief mass is added to those elements obtained by (8). Conflicting mass is redistributed to all elements within that area proportional to their assignment based on PCR6. This proportionality is applied to ensure that initial differences between elements are kept after this redistribution. If e.g., two elements were assigned *0.01* and *0.02* belief mass and *0.1* is to be redistributed to them. If this would not be done proportionally it would result in *0.06* and *0.08* belief mass. Since this is a distortion of the original difference in belief assignment we redistribute it proportionally, obtaining *0.043* and *0.086*. In this way, the fact that one element was assigned twice as much belief as the other is maintained after applying UPR.

$$m_c^{\text{upr}}(X) \ = \ m_c(X) \ + \sum_{\substack{X \cap X_j^* = X \\ \forall X_j \in \varnothing_{\mathcal{U}}}} \mathcal{N}_{\ell,j}(X) \cdot \mathcal{K}(X_j) \quad (9)$$

The redistribution is done $\forall X \in \varnothing_{\mathcal{U}}$ and we therefore obtain (9) for the UPR rule which is defined $\forall X \in D^{\Theta} \backslash \varnothing_{\mathcal{U}}$. In (9), $\mathcal{N}_{\ell,j}$ denotes the factor used to redistribute belief proportionally to element X_ℓ from X_j. I.e., it redistributes belief proportional to the original assignment made on the element compared to the total amount of belief assigned to all labels where belief is redistributed to (10).

$$\mathcal{N}_{\ell,j}(X) = \frac{m_c(X)}{\sum\limits_{\substack{X \cap X_j^* = X \\ X \notin \varnothing_{\mathcal{U}}}} m_c(X)} \quad (10)$$

V. Examples

For the two illustrative numerical examples in this section we consider the joined model as shown in Fig. 4. This model is the resulting Venn diagram when the world models of the different agents as well as the user are combined. For the example we have chosen a generic model with *4* specificity levels with various full parents and bridges. Problems that may be modelled in this manner are e.g., multi-label classification tasks, intent-recognition where similar actions may

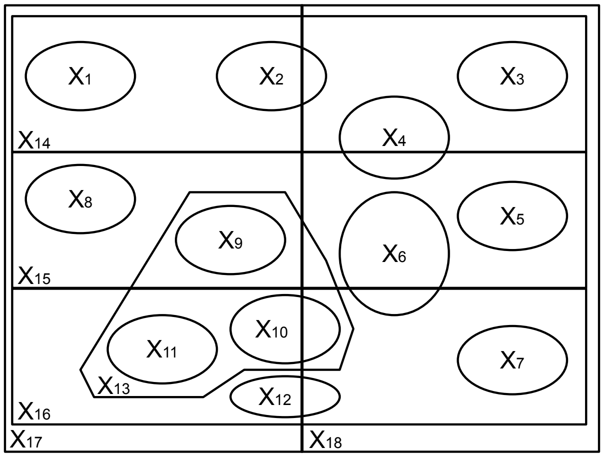


Figure 4. Venn diagram of a generic joined world model

point to different intentions, and crisis response. The focus of the examples is on negative confirmation since the positive confirmation is dealt with in the combination rule itself (the user's belief is treated the same way as an agent's by the interaction agent).

A. Example 1

For the first example the operator gives a negative confirmation on element X_{17}. In BCR terms: the truth is in X_{18} since $X_{17} \cup X_{18}$ encloses the entire frame of discernment and both elements are mutually exclusive. For BCR the mass from the dark coloured parts is then redistributed to all light grey coloured parts in Fig. 5(a). The set D_1 in BCR contains all elements that are enclosed in this entire light grey coloured part of the Venn Diagram. Some ambiguity however exists for elements like X_1, this element alone should be in D_2 since it has no common part elements involved in D_1. However, one could argue that since $X_1 = X_1 \cap X_{14}$ and since X_{14} is involved in D_1 — $X_{18} \cap X_{14} \in D_1$ — this should be an element in D_3. This shows the constraints of using BCR which is not able to deal with solution spaces with fully overlapping labels as already stated in Section III-C.

Besides the questions on how to construct the disjoint sets required for BCR, a choice has to be made on which specific BCR will be used. A possible solution would be to choose BCR1 since this rule treats elements from D_2 and D_3 exactly the same solving the ambiguity between these sets in our classification model. Using this rule, we redistribute the summed mass from the dark coloured region proportionally to all light grey coloured parts.

The same colour scheme is used to indicate what masses are set to zero and where the mass is redistributed to. A big difference is that UPR only constrains full child elements of X_{17}. This means that $\varnothing_{\mathcal{U}} = \{X_1, X_5, X_9, X_{11}, X_{17}\}$ and the UPR can be applied directly. Results from BCR1 and UPR for this example are shown in table I where $m_c(.)$ denotes the combined belief (by e.g., PCR6) before negative user confirmation. We see that UPR produces quite different results than BCR. Where BCR1 now assigns most belief to the specific label X_6, the UPR spreads the belief assignments over more generic labels. Based on the available information, this is considered preferable.

B. Example 2

Again in the example of Fig. 4, assume that the operator indicates that the object under consideration is not X_9. Using the dark colour for constrained parts in the Venn Diagram and light grey for the parts where mass is distributed to, we obtain Fig. 5(c) for BCR and Fig. 5(d) for UPR.

For BCR, D_1 now contains all parts of the Venn diagram with exception of X_9. Thus, the belief assigned to this part is proportionally redistributed to all other parts according to BCR1. In this example, the same ambiguity exists on whether X_9 should be in D_2 or D_3 as was the case in example *1*.

Using UPR however, the mass is redistributed to the smallest parent element and its child elements according to equations (8) and (9). For both these methods the results are shown in table II. Here, the UPR seems to focus the redistributed mass better than BCR. The latter simply redistributes the mass to all other elements that originally received a non-zero mass, whereas UPR only assigns it to the smallest full parent and its siblings.

Although a choice for a different BCR could influence this, we note that in BCR the decision to redistribute mass to an element is made based on the DSm cardinality where in UPR this decision is based on the relevance to the element that received a negative confirmation.

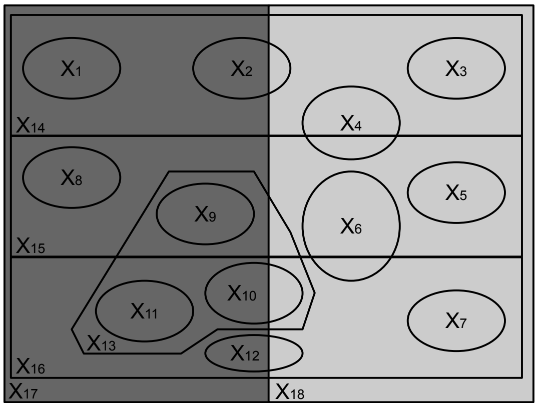
More examples have been considered but are omitted from this paper and only two illustrative examples are presented.

Table I
RESULTS WHEN X_{17} RECEIVES NEGATIVE CONFIRMATION

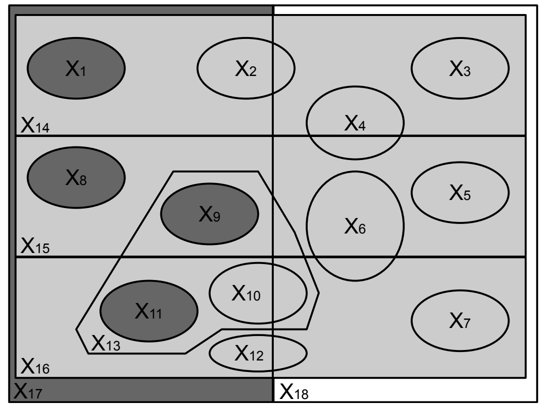
i	$m_c(X_i)$	$m_c^{\text{bcr1}}(X_i)$	$m_c^{\text{upr}}(X_i)$
6	*0.1*	*0.666*	*0.122*
8	*0.1*	*0*	*0*
9	*0.2*	*0*	*0*
10	*0.1*	*0*	*0.122*
11	*0.1*	*0*	*0*
13	*0.15*	*0*	*0.483*
15	*0.1*	*0*	*0.222*
17	*0.1*	*0*	*0*
18	*0.05*	*0.333*	*0.05*

Table II
RESULTS WHEN X_9 RECEIVES NEGATIVE CONFIRMATION

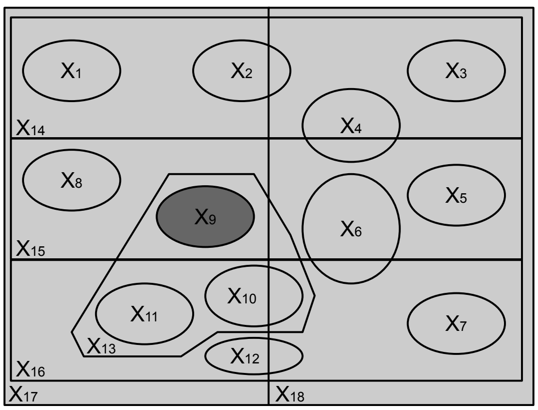
i	$m_c(X_i)$	$m_c^{\text{bcr1}}(X_i)$	$m_c^{\text{upr}}(X_i)$
6	*0.1*	*0.125*	*0.1*
8	*0.1*	*0.125*	*0.1*
9	*0.2*	*0*	*0*
10	*0.1*	*0.125*	*0.1571*
11	*0.1*	*0.125*	*0.1571*
13	*0.15*	*0.1875*	*0.2358*
15	*0.1*	*0.125*	*0.1*
17	*0.1*	*0.125*	*0.1*
18	*0.05*	*0.0625*	*0.05*



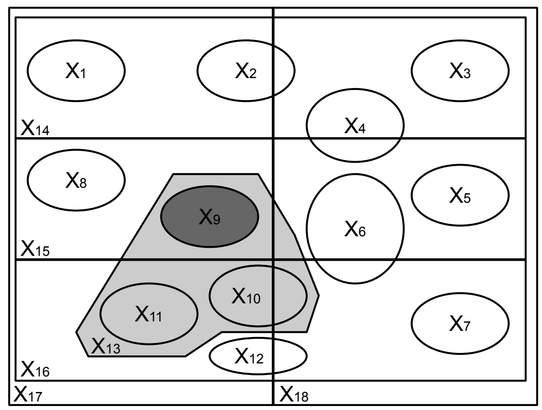
(a) Negative confirmation on X_{17} for BCR



(b) Negative confirmation on X_{17} for UPR



(c) Negative confirmation on X_9 for BCR



(d) Negative confirmation on X_9 for UPR

Figure 5. Venn diagrams in which the negative confirmation is dark grey and mass is redistributed to the light grey area

Future work will focus on more tests with operators using real data.

VI. Conclusions

When multiple agents need to solve a large and complex problem, an amount of disagreement inevitable when they have different world models and expertise. The involvement of an user in the MAS makes it even more complex. A joined hierarchical world model provides a solid framework for all parties involved in the problem solving to express their belief stated in an uniform way. Furthermore, this model may be expended by either the agents or the user during run-time.

We presented a new mechanism for the user to eliminate certain parts of the solution space by negative confirmation while the system keeps track of the conflict this produces in the background. This new UPR redistribution scheme uses knowledge about the solution space to transfer belief. This in contrast to e.g., BCR that mostly utilises (DSm) cardinality. Furthermore, the new UPR is based on subsets that may be constructed off-line since they are independent of which element needs to be constrained. This means a performance gain computationally compared to BCR where the three disjoint sets are constructed based on where the truth is. For real-time systems this performance gain is essential.

Based on a numerical example we showed that the new UPR assigns mass to the elements closest to the constrained element. With BCR, masses are transferred based on the cardinality without utilising the specific model knowledge that is available.

References

[1] Y. Yang, "An evaluation of statistical approaches to text categorization," *Information Retrieval*, vol. 1, no. 1-2, pp. 69–90, 1999.

[2] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor, "Kernel-based learning of hierarchical multilabel classification models," *Journal of Machine Learning Research*, vol. 7, pp. 1601–1626, 2006.

[3] F. Kang, R. Jin, and R. Sukthankar, "Correlated label propagation with application to multi-label learning," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1719–1726.

[4] S. Mukhopadhyay, S. Peng, R. Raje, M. Palakal, and J. Mostafa, "Multi-agent information classification using dynamic acquaintance lists," *Journal of the American Society for Information Science and Technology*, vol. 54, no. 10, pp. 966–975, May 2003.

[5] A. Kinie, M. L. L. Ndiaye, J.-J. Montois, and Y. Jacquelet, "A new multi criteria classification approach in a multi agent system applied to SEEG analysis," in *Proceedings of the IEEE Engineering in Medicine and Biology Society*, Lyon, France, 22–26 August 2007, pp. 2019–2022.

[6] J. R. Quinlan, *C4.5: Programs for machine learning*. San Francisco (CA) USA: Morgan Kaufmann Publishers Inc., 1993.

[7] M. E. Taylor, C. Matuszek, B. Klimt, and M. Witbrock, "Autonomous classification of knowledge into an ontology," in *Proceedings of the 20th International FLAIRS Conference*, May 2007.

[8] W. L. van Norden, F. Bolderheij, and C. M. Jonker, "Combining system and user belief on classification using the DSmT," in *Proceedings of the 11th International Conference on Information Fusion*, 30 June – 3 July 2008, pp. 768–775.

[9] B. Davey and H. Priestley, *Introduction to Lattices and Order*. Cambridge University Press, 2002.

[10] F. Smarandache and J. Dezert, "An introduction to the DSm theory for the combination of paradoxical, uncertain and imprecise sources of information," in *Proceedings of the 13th International Congress of Cybernetics and Systems*, 6–10 July 2005.

[11] J. Dezert and F. Smarandache, *Advances and Applictaions of the DSmT (collected works)*. Rehoboth (MA), USA: American Research Press, 2004, vol. 1, ch. 3. Partial ordering on hyper-power sets.

[12] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni, "Incremental algorithms for hierarchical classification," *Journal of Machine Learning Research*, vol. 7, pp. 31–54, 2006.

[13] B. Mertens, "Reasoning with uncertainty in the situational awareness of air targets," Master's thesis, Delft University of Technology, Delft, 2004.

[14] A. P. Dempster, "A generalization of bayesian inference," *Journal of the Royal Statistical Society*, vol. 30, no. Series B, pp. 205–247, 1968.

[15] G. Shafer, *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[16] A. Martin and C. Oswald, *Advances and Applications of DSmT for Information Fusion (collected works)*. Rehoboth (MA): American Research Press, 2006, vol. 2, ch. 2. A new generalization of the proportional conflict redistribution rule stable in terms of decision, pp. 69–88.

[17] F. Smarandache and J. Dezert, *Advances and Applications of DSmT for Information Fusion (collected works)*. Rehoboth, USA: American Research Press, 2006, vol. 2, ch. 9. Belief Conditioning Rules, pp. 237–268.