Merging Uncertain Information with Semantic Heterogeneity in XML

Anthony Hunter[*] and Weiru Liu[†]

March 11, 2005

Abstract

Semi-structured information in XML can be merged in a logic-based framework [Hun02, Hun02b]. This framework has been extended to deal with uncertainty, in the form of probability values, degrees of beliefs, or necessity measures, associated with leaves (i.e., textentries) in the XML documents [HL04a]. In this paper we further extend this approach to modelling and merging uncertain information that is defined at different levels of granularity of XML textentries, and to modelling and reasoning with XML documents that contain semantically heterogeneous uncertain information on more complex elements in XML subtrees. We present the formal definitions for modelling, propagating and merging semantically heterogeneous uncertain information and explain how they can be handled using logic-based fusion techniques.

1 Introduction

With XML fast emerging as the dominant standard for representing and exchanging information over the web, the need for modelling uncertainty in the information has begun to be addressed. In [NJ02], a probabilistic approach is taken to model and reason with uncertain information at different levels of tags in a single XML document. The final probability of the value of a specific tag is calculated via multiple conditional probabilities on its ancesters' tags. In another approach [KKA05], probability values are also attached to tags, but it requires that the probabilities of a set of values associated with a single tag must sum to 1.0, a condition that was not required in [NJ02]. A simple merging method is also provided to integrate two probabilistic XML trees in [KKA05], whilst [NJ02] did not consider multiple XML documents. Since [KKA05] does not use much of the background knowledge to verify the probabilistic XML documents before merging, even two simple XML files as input can produce a huge number of possible XML documents as output (see Conclusion for details), which makes the method difficult to use in practice.

In contrast, our approach to modelling, reasoning, and merging XML documents with uncertain information ([HL04a]) concerns information within the logical fusion framework [HS04] where background knowledge can provide additional information to facilitate merging and reduce redundancy and inconsistency among information. In this paper, we focus on structured reports. The format of a structured report is an XML document where the tagnames provide the semantic structure and coherence to the document and the textentries (i.e. leaves) are restricted to (1) individual words or simple phrases from a scientific nomenclature/terminology and (2) individual numerical values with units. For instance, a structured report on deposits of a particular underground location can be represented using the tagnames `deposit` with textentries such as `water`, `oil`, `gas`, and `sand`, etc.

[*] Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK

[†] School of Computer Science, Queen's University Belfast, Belfast, Co Antrim BT7 1NN, UK

Example 1 *Consider the following two structured reports which are for the same area being explored. Both of them define a mass function on the textentry* `deposit`.

```
⟨report⟩
 ⟨source⟩ Experiment1 ⟨/source⟩
 ⟨date⟩ 19/3/02 ⟨/date⟩
 ⟨location⟩ NorthSea ⟨/location⟩
 ⟨layer⟩ layer 7 : 100m − 120m ⟨/layer⟩
 ⟨deposit⟩
  ⟨belfunction⟩
   ⟨mass value = “0.4”⟩
    ⟨massitem⟩water⟨/massitem⟩
    ⟨massitem⟩oil⟨/massitem⟩
   ⟨/mass⟩
   ⟨mass value = “0.6”⟩
    ⟨massitem⟩gas⟨/massitem⟩
   ⟨/mass⟩
  ⟨/belfunction⟩
 ⟨/deposit⟩
⟨/report⟩
```

```
⟨report⟩
 ⟨source⟩ Experiment2 ⟨/source⟩
 ⟨date⟩ 19 March 2002 ⟨/date⟩
 ⟨location⟩ NorthSea ⟨/location⟩
 ⟨layer⟩ layer 7 : 100m − 120m⟨/layer⟩
 ⟨deposit⟩
  ⟨belfunction⟩
   ⟨mass value = “0.2”⟩
    ⟨massitem⟩water⟨/massitem⟩
   ⟨/mass⟩
   ⟨mass value = “0.8”⟩
    ⟨massitem⟩gas⟨/massitem⟩
   ⟨/mass⟩
  ⟨/belfunction⟩
 ⟨/deposit⟩
⟨/report⟩
```

Let τ_1, τ_2 be two logical terms that represent the two XML documents above, and let X be a variable. A fusion predicate $\texttt{Dempster}(\tau_1, \tau_2, X)$ *defined later in Section 2 takes these two XML documents as inputs and generates a merged structured report that grounds X with the combined mass function segment as shown below.*

```
⟨report⟩
 ⟨source⟩ Exp1 and Exp2 ⟨/source⟩
 ⟨date⟩ 19/3/02 ⟨/date⟩
 ⟨location⟩ NorthSea ⟨/location⟩
 ⟨layer⟩ layer 7 : 100m − 120m ⟨/layer⟩
 ⟨deposit⟩
  ⟨belfunction⟩
   ⟨mass value = “0.143”⟩
    ⟨massitem⟩water⟨/massitem⟩
   ⟨/mass⟩
   ⟨mass value = “0.857”⟩
    ⟨massitem⟩gas⟨/massitem⟩
   ⟨/mass⟩
  ⟨/belfunction⟩
 ⟨/deposit⟩
⟨/report⟩
```

In our approach, each structured report can isomorphically be represented as a logical term: Each tagname is a function symbol, and each textentry is a constant symbol. Furthermore, subtrees of a structured report can be isomorphically represented as subterms in logic. In this way, the information in each structured report can be captured in a logical language. We have also defined a range of predicates, in a Prolog knowledgebase, that capture useful relationships between structured reports, and so a set of them can then be analysed or merged as Prolog queries to a Prolog knowledgebase. In this way, a query to merge some structured reports can be handled by recursive calls to Prolog to merge the subtrees in the structured reports. This gives a context-dependent logic-based approach to merging that is sensitive to the uncertain information in the structured reports and to the background knowledge in the Prolog knowledgebase.

In [HL04a], a method to model and merge uncertain information, represented by probabilities, mass functions in the Dempster-Shafer theory of evidence (DS theory) [Sha76] and necessity measures in possibility theory [DP88], was proposed. Example 1 illustrates how a mass function can be encoded into XML format and how two mass functions on the same set of values can be merged to produce a combined XML document. Details of the formal definition and merging procedure will be reviewed in Section 2.

Here in this and subsequent examples, we use some simplified data from the petroleum exploration domain. The main purpose of petroleum exploration is to analyse qualitatively and calculate quantitatively the well logging data in order to predict the possible deposits in particular locations. The well logging data are digital records which can reflect the underground physical features, for instance, electronic resistance, micro-electrode resistance, natural gamma ray, etc. They are collected by well logging equipment inside the well from the ground level to some depth underground. The whole depth from the ground level to the bottom of the well is divided into layers (such as, 100meters to 150meters) based on the digital data collected and the values of these physical features can give indications of layers with possible deposits. The first two XML documents in Example 1 show how an expert can predict a possible deposit of a particular layer, by examining the digital data of the layer. Since equipment used is subject to noise and inaccuracy, multiple experiments are needed in order to make an accurate prediction. Furthermore, the general analysis of the broader area of the physical features of the location often provides some additional information for predication. This knowledge can equally be represented as XML documents and be used to assist predication when necessary.

The main focus of [HL04a] is the modelling and merging of uncertain information associated with textentries in XML documents. Multiple pieces of uncertain information concerning the same issue (such as `deposit` in the above example) are assumed to be specified on the same set of possible values. However, [HL04a] does not consider situations where one piece of information uses more specific values than another nor the situation where one piece of information is described on one set of values and another is on a different set of values where these two sets of values are inter-connected.

We elaborate this issue further here. Assume that for a targeted layer of a specific well of a particular area, we only wish to conclude whether the layer contains either solid or liquid materials, regardless of the details of the substance. Then we use a set of values $\{\texttt{solid}, \texttt{liquid}\}$ to bear any information we have about the layer. However, we could make this information more specific by giving different types of solid and liquid substances, such as, *stone, sand, water, gas, oil*. Therefore some uncertain information can be described on this detailed set of values $\{\texttt{stone}, \texttt{sand}, \texttt{water}, \texttt{gas}, \texttt{oil}\}$. This latter set of values has a finer granularity than the former one. Furthermore, since possible deposits of a layer are often drawn through interpreting well logging data other than being observed directly, well logging data will directly influence the prediction. For instance, it is commonly known that a set of data is first interpreted in terms of geographical features, and then the assumed features are used to predict possible deposits. In this situation, the information is represented on one set of values (geographical features, e.g., lithology) and the conclusion is on another set (e.g., deposit). The information from the given set of values should be propagated to the destination set of values as a new distribution of beliefs. To deal with these situations, in this paper, we extend our the approach to merging multiple pieces of uncertain information where

- evidence is specified at different levels of granularity on the same concept as textentries. We refer to two pieces of this type of evidence as *semantically homogeneous*. In this case, a value in a coarser set can be replace by a set of values in a finer set. The example above relating *solid* and *liquid* with *stone, sand, water, gas*, and *oil*, belongs to this category.

- evidence is specified on inter-related concepts as textentries. We refer to two pieces of this type of evidence as *semantically heterogeneous*. Example 3 below relating *stone*, *sand*, *water*, *gas*, and *oil*, with lithologies *L1, L2* etc. belongs to this category.

- evidence is assigned to heterogeneous subtrees involving multiple concepts. We also refer to two pieces of this type of evidence as *semantically heterogeneous*. For instance, if we have a set of values measuring the *lithology* of a layer and another set evaluating the type of *deposit* of the layer, and we would like to know both the *lithology* and the *deposit* of the layer, then the joint set from these two sets says what lithology and what type of deposit a location has.

The first two types of evidence are illustrated by Examples 2 and 3 respectively and the third type of uncertain information is demonstrated by Example 4.

Example 2 *Consider the two structured reports about a specific underground layer. The first report gives more precise descriptions of the possible deposit under a particular layer with probabilities whilst the other gives a more general suggestion of the possible deposit. These two reports describe the same problem with different levels of abstraction (different granularities), so they have uncertain information that is* semantically homogeneous.

```
⟨report⟩                                   ⟨report⟩
 ⟨deposit⟩                                  ⟨deposit⟩
  ⟨probability⟩                              ⟨probability⟩
   ⟨prob value = “0.2”⟩water⟨/prob⟩           ⟨prob value = “0.4”⟩liquid⟨/prob⟩
   ⟨prob value = “0.8”⟩sand⟨/prob⟩            ⟨prob value = “0.6”⟩solid⟨/prob⟩
  ⟨/probability⟩                             ⟨/probability⟩
 ⟨/deposit⟩                                 ⟨/deposit⟩
⟨/report⟩                                  ⟨/report⟩
```

Evidence bearing on a finer granularity (e.g., `deposit` *with values* `water`, `gas` *etc) would have impact on a coarser granularity (e.g.,* `deposit` *with values* `liquid`, `solid` *etc) or vice versa. It is sensible to consider both pieces of evidence at the same level of granularity if one piece of evidence can be propagated to the level of the other. This is the first topic we will look into in this paper.*

Example 3 *The following two structured reports provide two different but inter-related pieces of evidence about the same layer of the same well. The evidence in the left-hand XML document reports directly on the potential physical nature of the deposit. This is commonly used for prediction and this information can come from the general knowledge about the area. Whilst the second XML document reports on the observations in terms of lithology made by the equipment. From the lithological features, we can determine the physical nature of the deposit (or vice versa). To make use of this second XML report in prediction, we need to have a proper mapping function which specifies how the interpretations of lithology imply deposits, and then both of these reports can be merged. Since these two reports provide uncertain information on two different but inter-related concepts i.e.,* `deposit` *and* `lithology`*, we refer to them as* semantically heterogeneous. *Propagating a piece of uncertain information from one set of values to a different set of values is the second topic we will investigate in this paper.*

```
⟨report⟩                              ⟨report⟩
 ⟨deposit⟩                             ⟨lithology⟩
  ⟨belfunction⟩                         ⟨belfunction⟩
   ⟨mass value = “0.2”⟩                  ⟨mass value = “0.3”⟩
    ⟨massitem⟩water⟨/massitem⟩            ⟨massitem⟩L1⟨/massitem⟩
    ⟨massitem⟩oil⟨/massitem⟩              ⟨massitem⟩L3⟨/massitem⟩
   ⟨/mass⟩                               ⟨/mass⟩
   ⟨mass value = “0.8”⟩                  ⟨mass value = “0.7”⟩
    ⟨massitem⟩gas⟨/massitem⟩              ⟨massitem⟩L2⟨/massitem⟩
   ⟨/mass⟩                               ⟨/mass⟩
  ⟨/belfunction⟩                        ⟨/belfunction⟩
 ⟨/deposit⟩                            ⟨/lithology⟩
⟨/report⟩                             ⟨/report⟩
```

Example 4 *Consider the following two structured reports which again are for the same layer of the same well. In the left report, there are two probability distributions on two textentries respectively. When we use this information to make a prediction, we can either use the information about the* `deposit` *or* `lithology` *since the former may have been derived from the later or vice versa. Whilst in the right report, the child of the* ⟨prob value = "..."⟩ *tag is not a textentry, it is in fact a subtree involving two concepts* `deposit` *and* `lithology`*. This information can be the summary of general knowledge about this area saying what deposit is associated with what lithologies. For the purpose of prediction, uncertainties assigned to the pairs of values (e.g.,* (water, L1)*) have to be re-assigned to values of* `deposit` *such as* water, oil *etc. Following this uncertainty re-assignment, the newly derived uncertain information on* `deposit` *can be merged with the information in the left XML. These two pieces of uncertain information are also referred to as* semantically heterogeneous, *however, they require a different method to propagate before they can be merged. Subtree uncertain information is the third topic we will study in this paper.*

```
⟨report⟩                                  ⟨report⟩
  ⟨source⟩ experiment3 ⟨/source⟩            ⟨source⟩ General knowledge ⟨/source⟩
  ⟨date⟩ 19/3/02 ⟨/date⟩                    ⟨date⟩ 19 March 2002 ⟨/date⟩
  ⟨location⟩ NorthSea ⟨/location⟩           ⟨location⟩ NorthSea ⟨/location⟩
  ⟨layer⟩ 150m − 155m ⟨/layer⟩              ⟨date⟩ 150m − 160m ⟨/layer⟩
  ⟨deposit⟩                                 ⟨probability⟩
    ⟨probability⟩                             ⟨prob value = "0.4"⟩
      ⟨prob value = "0.2"⟩water⟨/prob⟩          ⟨deposit⟩water⟨/deposit⟩
      ⟨prob value = "0.8"⟩gas⟨/prob⟩            ⟨lithology⟩L1⟨/lithology⟩
    ⟨/probability⟩                            ⟨/prob⟩
  ⟨/deposit⟩                                  ⟨prob value"0.6"⟩
  ⟨lithology⟩                                   ⟨deposit⟩gas⟨/deposit⟩
    ⟨probability⟩                               ⟨lithology⟩L2⟨/lithology⟩
      ⟨prob value = "0.3"⟩L1⟨/prob⟩           ⟨/prob⟩
      ⟨prob value = "0.7"⟩L2⟨/prob⟩         ⟨/probability⟩
    ⟨/probability⟩                        ⟨/report⟩
  ⟨/lithology⟩
⟨/report⟩
```

So the purpose of this paper is to significantly extend our previous paper on handling uncertainty [HL04a] by presenting techniques for merging structured reports with uncertainty expressed: (1) at different levels of granularity; (2) on different but inter-related sets of values; and (3) on subtrees. We will proceed as follows. In Section 2, we present formal definitions of logical representations of XML documents, review the basics of DS theory, and provide formal definitions of modelling and merging uncertain information in structured reports in the form of mass functions on the same textentry of two XML documents. In Section 3, we consider propagating and merging uncertain information at different levels of granularity. In Section 4, we investigate methods of reasoning with semantically heterogeneous uncertain information on subtrees. In Section 5, we compare our work with related research. Finally, in Section 6 we provide conclusions.

2 Structured reports

We now briefly review definitions for structured reports, Dempster-Shafer theory of evidence (DS theory), for representing uncertain information in structured reports.

2.1 Basic definitions

Each structured report is an XML document, but not vice versa, as defined below. This restriction means that we can easily represent each structured report by a ground term in classical logic.

Definition 1 Structured report: *If φ is a tagname (i.e an element name), and ϕ is a textentry, then $\langle\varphi\rangle\phi\langle/\varphi\rangle$ is a structured report. If φ is a tagname (i.e an element name), ϕ is a textentry, θ is an attribute name, and κ is an attribute value, then $\langle\varphi\ \theta = \kappa\rangle\phi\langle/\varphi\rangle$ is a structured report. If φ is a tagname and $\sigma_1, ..., \sigma_n$ are structured reports, then $\langle\varphi\rangle\sigma_1...\sigma_n\langle/\varphi\rangle$ is a structured report.*

The definition for a structured report is very general. In practice, we would expect a DTD for a given domain. For instance, we would expect that for an implemented system that merges petroleum exploration reports, there would be a corresponding DTD. One of the roles of a DTD, say for petroleum exploration reports, would be to specify the minimum constellation of tags that would be expected of a petroleum exploration report. We may also expect integrity constraints represented in classical logic to further restrict appropriate structured reports for a domain [HS04]. In this paper, we will impose some further constraints on structured reports, in Section 2.3, to support the handling of uncertainty.

Clearly each structured report is isomorphic to a tree with the non-leaf nodes being the tagnames and the leaf nodes being the textentries. When we refer to a subtree (of a structured report), we mean a subtree formed from the tree representation of the structured report, where the root of the subtree is a tagname and the leaves are textentries. We formalize this as follows.

Definition 2 Subtree: *Let σ be a structured report and let ρ be a tree that is isomorphic to σ. A tree ρ' is a subtree of ρ iff (1) the set of nodes in ρ' is a subset of the set of nodes in ρ, and (2) for each node φ_i in ρ', if φ_i is the parent of φ_j in ρ, then φ_j is in ρ' and φ_i is the parent of φ_j in ρ'. By extension, if σ' is a structured report, and ρ' is isomorphic to σ', then we describe σ' as a subtree of σ.*

Each structured report is also isomorphic with a ground term (of classical logic) where each tagname is a function symbol and each textentry is a constant symbol.

Definition 3 Abstract term: *Each structured report is isomorphic with a ground term (of classical logic) called an abstract term. This isomorphism is defined inductively as follows: (1) If $\langle\varphi\rangle\phi\langle/\varphi\rangle$ is a structured report, where ϕ is a textentry, then $\varphi(\phi)$ is an abstract term that is isomorphic with $\langle\varphi\rangle\phi\langle/\varphi\rangle$; (2) If $\langle\varphi\ \theta = \kappa\rangle\phi\langle/\varphi\rangle$ is a structured report, where ϕ is a textentry, then $\varphi(\phi, \kappa)$ is an abstract term that is isomorphic with $\langle\varphi\ \theta = \kappa\rangle\phi\langle/\varphi\rangle$; and (3) If $\langle\varphi\rangle\phi_1..\phi_n\langle/\varphi\rangle$ is a structured report, and ϕ'_1 is an abstract term that is isomorphic with ϕ_1, ..., and ϕ'_n is an abstract term that is isomorphic with ϕ_n, then $\varphi(\phi'_1, .., \phi'_n)$ is an abstract term that is isomorphic with $\langle\varphi\rangle\phi_1..\phi_n\langle/\varphi\rangle$.*

Via this isomorphic relationship, we can refer to a branch of an abstract term by using the branch of the isomorphic structured report, and we can refer to a subtree of an abstract term by using the subtree of the isomorphic structured report. Note, Definition 1 describes how an XML document can be defined recursively starting from the simplist one which has only one tagname and one value associated with the tagname. Also Definition 3 specifies how a tree structure like XML document can be equally described as a logical term which also reflects the relationships between tagnames and their values. For instance, XML information $\langle$`date`$\rangle$`03/03/99`$\langle/$`date`$\rangle$ is denoted as `date`(`03/03/99`) in logics where `03/03/99` can be understood as the value of attribute `date`.

Example 5 *Consider the following structured report.*

$$\begin{array}{l}\langle\texttt{fieldreport}\rangle\\ \quad\langle\texttt{log}\rangle\langle\texttt{deposit}\rangle\texttt{liquid}\langle/\texttt{deposit}\rangle\langle\texttt{lithology}\rangle\texttt{L1}\langle/\texttt{lithology}\rangle\langle/\texttt{log}\rangle\\ \quad\langle\texttt{layer}\rangle\texttt{250m}-\texttt{300m}\langle/\texttt{layer}\rangle\\ \langle/\texttt{fieldreport}\rangle\end{array}$$

This can be represented by the following abstract term:

$$\texttt{fieldreport}(\texttt{log}(\texttt{deposit}(\texttt{liquid}),\texttt{lithology}(\texttt{L1})),\texttt{layer}(\texttt{250m}-\texttt{300m}))$$

In this abstract term, `fieldreport/log/deposit` *is a branch.*

2.2 Basics of Dempster-Shafer Theory of Evidence

The Dempster-Shafer theory (DS theory) of evidence provides a mechanism for modelling and reasoning with uncertain information in a numerical way, especially when it is not possible to assign a proportion of the total belief to single elements of a set of values. DS theory ([Sha76, Sme88]) has a commonly accepted advantage over probability theory in terms of assigning a proportion of an agent's belief to a subset of a set of possible values rather than only on singletons, and assigning any unspecified proportion to the whole set. This is especially useful when the evidence supporting an agent's belief is not accurate or incomplete. Furthermore, multiple pieces of evidence can be accumulated over time on the same subject and these pieces of evidence can be combined/merged in some way in order to draw a conclusion out of them. Dempster's combination rule in DS theory provides a simple mechanism to achieve this objective. Due to these two advantages provided by DS theory, we have chosen it to model, reason and merge uncertain information in structured reports.

Let Ω be a finite set containing mutually exclusive and exhaustive solutions to a question. Ω is called the **frame of discernment**. A **mass function**, also called a **basic probability assignment**, captures the impact of a piece of evidence on subsets of Ω. A mass functions $m : \wp(\Omega) \to [0, 1]$ satisfies:

$$(1)\ m(\emptyset) = 0$$

$$(2)\ \Sigma_{A\subseteq\Omega}\ m(A) = 1$$

When $m(A) > 0$, A is referred to as a **focal element**. To obtain the total belief in a subset A, i.e. the extent to which all available evidence supports A, we need to sum all the mass assigned to all subsets of A. A **belief function**, $Bel : \wp(\Omega) \to [0, 1]$, is defined as

$$Bel(A) = \Sigma_{B\subseteq A} m(B)$$

A **plausibility function**, denoted $Pl : \wp(\Omega) \to [0, 1]$, is defined as

$$Pl(A) = 1 - Bel(\bar{A}) = \Sigma_{B\cap A\neq\emptyset}\ m(B)$$

Dempster's rule of combination below shows how two mass functions m_1 and m_2 on the same frame of discernment from independent sources, can be combined to produce a merged mass function.

$$m_1 \oplus m_2(C) = \frac{\Sigma_{A\cap B=C}\ (m_1(A) \times m_2(B))}{1 - \Sigma_{A\cap B=\emptyset}\ (m_1(A) \times m_2(B))}$$

A mass function reduces to a probability distribution when every focal element is in fact a singleton. It is with this aspect that in this paper, we view probability theory as a special case of DS theory.

2.3 Representing uncertain information

In order to support the representation of uncertain information in structured reports, we need some further formalization. First, we assume a set of tagnames that are reserved for representing uncertain information. Second, we assume some constraints on the use of these tags so that we can ensure they are used in a meaningful way with respect to probability theory and Dempster-Shafer theory of evidence. The set of **key uncertainty tagnames** for this paper are `probability` and `belfunction`. The set of **subsidiary uncertainty tagnames** for this paper are `prob`, `multiitem`, `mass`, and `massitem`. The union of the key uncertainty tagnames and the subsidiary uncertainty tagnames is the set of **reserved tagnames**.

Definition 4 *([HL04a]) The structured report* $\langle\texttt{probability}\rangle\sigma_1,..,\sigma_n\langle/\texttt{probability}\rangle$ *is called a* **probability-valid component (ProVC)** *iff each* $\sigma_i \in \{\sigma_1,..,\sigma_n\}$ *is of the form* $\langle\texttt{prob value} = \kappa\rangle\phi\langle/\texttt{prob}\rangle$ *where* $\kappa \in [0,1]$ *and* ϕ *is a textentry.*

All textentries ϕ_i between $\langle\texttt{prob value} = \kappa_\texttt{i}\rangle\phi_i\langle/\texttt{prob}\rangle$ are elements of a pre-defined set containing mutually exclusive and exhaustive values that the related tagname can take.

Example 6 *The following is a ProVC which corresponds to a probability distribution* $p(\texttt{water}) = 0.2$ *and* $p(\texttt{gas}) = 0.8$.

$\langle\texttt{probability}\rangle$
$\quad\langle\texttt{prob value} = \text{“0.2”}\rangle\texttt{water}\langle/\texttt{prob}\rangle$
$\quad\langle\texttt{prob value} = \text{“0.8”}\rangle\texttt{gas}\langle/\texttt{prob}\rangle$
$\langle/\texttt{probability}\rangle$

Definition 5 *The structured report* $\langle\texttt{probability}\rangle\sigma_1,..,\sigma_n\langle/\texttt{probability}\rangle$ *is called a* **subtree probability-valid component (ProSC)** *iff for each* $\sigma_i \in \{\sigma_1,..,\sigma_n\}$*,* σ_i *is of the form*

$$\langle\texttt{prob value} = \kappa_i\rangle\langle\texttt{multiitem}\rangle\sigma^i_1,...,\sigma^i_m\langle/\texttt{multiitem}\rangle\langle/\texttt{prob}\rangle$$

and for each $\sigma^i_j \in \{\sigma^i_1,..,\sigma^i_m\}$*,* σ^i_j *is of the form* $\langle\psi^i_{j_l}\rangle\phi^i_{j_l}\langle/\psi^i_{j_l}\rangle$*, and* $\kappa_i \in [0,1]$*,* $\psi^i_{j_l}$ *is a tagname, and* $\phi^i_{j_l}$ *is a textentry.*

Example 7 *The following is a ProSC that models a probability distribution on a compound set of values with* $p(\{\texttt{water},\texttt{L1}\}) = 0.4$ *and* $p(\{\texttt{gas},\texttt{L2}\}) = 0.6$.

$\langle\texttt{probability}\rangle$
$\quad\langle\texttt{prob value} = \text{“0.4”}\rangle$
$\quad\quad\langle\texttt{multiitem}\rangle$
$\quad\quad\quad\langle\texttt{deposit}\rangle\texttt{water}\langle/\texttt{deposit}\rangle$
$\quad\quad\quad\langle\texttt{lithology}\rangle\texttt{L1}\langle/\texttt{lithology}\rangle$
$\quad\quad\langle/\texttt{multiitem}\rangle$
$\quad\langle/\texttt{prob}\rangle$
$\quad\langle\texttt{prob value} = \text{“0.6”}\rangle$
$\quad\quad\langle\texttt{multiitem}\rangle$
$\quad\quad\quad\langle\texttt{deposit}\rangle\texttt{gas}\langle/\texttt{deposit}\rangle$
$\quad\quad\quad\langle\texttt{lithology}\rangle\texttt{L2}\langle/\texttt{lithology}\rangle$
$\quad\quad\langle/\texttt{multiitem}\rangle$
$\quad\langle/\texttt{prob}\rangle$
$\langle/\texttt{probability}\rangle$

The reserved tagname `multiitem` within tagname `prob` indicates that there are multiple concepts in this uncertain information. In the above example, each probability value is attached to a compound element combining `deposit` and `lithology`.

Definition 6 *([HL04a]) The structured report* $\langle\texttt{belfunction}\rangle\sigma_1,..,\sigma_n\langle/\texttt{belfunction}\rangle$ *is called a* **belfunction-valid component (BelVC)** *iff for each* $\sigma_i \in \{\sigma_1,..,\sigma_n\}$ σ_i *is of the form* $\langle\texttt{mass value} = \kappa_i\rangle\psi_i\langle/\texttt{mass}\rangle$ *and* ψ_i *is in the form*

$$\langle\texttt{massitem}\rangle\phi_{i_1}\langle/\texttt{massitem}\rangle,\ldots,\langle\texttt{massitem}\rangle\phi_{i_x}\langle/\texttt{massitem}\rangle$$

where $\kappa_i \in [0,1]$ *and* ϕ *is a textentry. To make the subsequent notation simpler, we also let* $\psi_i = \{\phi_{i_1},\ldots,\phi_{i_x}\}$*. In this way, a BelVC can be represented as a collection of (subset, mass value) pairs,* $(\psi_i,\kappa_i), i = 1,\ldots,n$.

Example 8 *The following is a BelVC on a single tagname* `deposit` *with* $m(\{\texttt{water},\texttt{oil}\}) = 0.2$ *and* $m(\{\texttt{gas}\}) = 0.8$.

```
⟨belfunction⟩
 ⟨mass value = “0.2”⟩
  ⟨massitem⟩water⟨/massitem⟩
  ⟨massitem⟩oil⟨/massitem⟩
 ⟨/mass⟩
 ⟨mass value = “0.8”⟩
  ⟨massitem⟩gas⟨/massitem⟩
 ⟨/mass⟩
⟨/belfunction⟩
```

The textentries in a BelVC are elements of a pre-defined set containing mutually exclusive and exhaustive values for the related tagname as in the case for ProVCs. We now provide the definition of mass functions on subtrees.

Definition 7 *The structured report* $\langle\texttt{belfunction}\rangle\sigma_1,..,\sigma_n\langle/\texttt{belfunction}\rangle$ *is called a* **subtree belfunction-valid component (BelSC)** *iff for each* $\sigma_i \in \{\sigma_1,..,\sigma_n\}$ σ_i *is of the form* $\langle\texttt{mass value} = \kappa_i\rangle\ \psi_i\ \langle/\texttt{mass}\rangle$ *and* ψ_i *is in the form*

$$\langle\texttt{multiitem}\rangle\varphi_{i_1}\langle/\texttt{multiitem}\rangle\ldots\langle\texttt{multiitem}\rangle\varphi_{i_x}\langle/\texttt{multiitem}\rangle$$

and each φ_{i_j} *in* $\{\varphi_{i_1},\ldots,\varphi_{i_x}\}$ *is in the form*

$$\langle\rho^i_{j_1}\rangle\phi^i_{j_1}\langle/\rho^i_{j_1}\rangle,\ldots,\langle\rho^i_{j_l}\rangle\phi^i_{j_l}\langle/\rho^i_{j_l}\rangle$$

where $\kappa_i \in [0,1]$, $\rho^i_{j_t}$ *are tagnames, and* $\phi^i_{j_t}$ *are textentries. Equally,* $\psi_i = \{<\phi^i_{1_1},\ldots,\phi^i_{1_p}>,\ldots,<\phi^i_{x_1},\ldots,\phi^i_{x_m}>\}$ *can be used to stand for a subset with mass value* κ_i *where the subset consists of elements with multiple atom values.*

Example 9 *The following is a BelSC providing a mass function on a subtree.*

```
⟨belfunction⟩
 ⟨mass value = “0.4”⟩
  ⟨multiitem⟩
   ⟨deposit⟩water⟨/deposit⟩
   ⟨lithology⟩L1⟨/lithology⟩
  ⟨/multiitem⟩
  ⟨multiitem⟩
   ⟨deposit⟩oil⟨/deposit⟩
   ⟨lithology⟩L3⟨/lithology⟩
  ⟨/multiitem⟩
 ⟨/mass⟩
 ⟨mass value = “0.6”⟩
  ⟨multiitem⟩
   ⟨deposit⟩gas⟨/deposit⟩
   ⟨lithology⟩L2⟨/lithology⟩
  ⟨/multiitem⟩
 ⟨/mass⟩
⟨/belfunction⟩
```

If a belief function is defined on a subtree, then for each mass value, its elements should come from different frames. So the tagnames should be distinct. In addition, if the subtree involves n tagnames, then in each ($\langle\texttt{multiitem}\rangle$, $\langle\texttt{/multiitem}\rangle$) pair, there should be n tagnames. These are the two constraints we impose on BelSCs. When a tagname among these n names is missing, this part of the XML can be extended to include the missing tagname. More specifically, if we are defining a mass function for a subtree involving frames Θ_1 and Θ_2, then for a mass assignment that involves elements from just one of the two frames, we can extend it to include all the elements in the other frame. For example, the mass function in Example 9 gives

$$m(\{<\texttt{water},\texttt{L1}>,<\texttt{oil},\texttt{L3}>\}) = 0.4,\ \ \texttt{m}(\{<\texttt{gas},\texttt{L2}>\}) = 0.6.$$

If it was the case that $m(\{<\texttt{gas},\texttt{L2}>\}) = 0.4$ is mis-represented as $m(\{\texttt{gas}\}) = 0.4$, then it can be extended into $m(\{<\texttt{gas},\texttt{L1}>,<\texttt{gas},\texttt{L2}>,...,<\texttt{gas},\texttt{L10}>\}) = 0.4$. This means gas is compatible with all the lithologies. Therefore, in the following, we always assume that a BelSC complies with these two constraints.

The ProVCs, ProSCs, BelVCs, and BelSCs are referred to as **uncertainty components** and are normally part of larger structured reports. Normally, we would expect that for an application, the DTD for the structured reports would exclude a key uncertainty tag as the root of a structured report. In other words, the key uncertainty tags are roots of subtrees nested within larger structured reports. We also assume various integrity constraints on the use of the uncertainty components.

Definition 8 *Let* $\langle\texttt{probability}\rangle\sigma_1,..,\sigma_n\langle\texttt{/probability}\rangle$ *be a ProVC or a ProSC, and let* $\sigma_i \in \{\sigma_1,..,\sigma_n\}$ *be either of the form* $\langle\texttt{prob value} = \kappa_i\rangle\phi_i\langle\texttt{/prob}\rangle$ *or of the form* $\langle\texttt{prob value} = \kappa_i\rangle$ $\langle\texttt{multiitem}\rangle$ $\phi_{i_1},\ldots,\phi_{i_l}$ $\langle\texttt{/multiitem}\rangle$ $\langle\texttt{/prob}\rangle$*. This component adheres to the* **full probability distribution constraint** *iff the following two conditions hold:*

(1) $\Sigma_i\kappa_i = 1$
(2) *for all* $i,j,$ *if* $1 \le i \le n$ *and* $1 \le j \le n$ *and* $i \ne j,$ *then* $\phi_i \ne \phi_j$ *or* $\{\phi_{i_1},\ldots,\phi_{i_l}\} \ne \{\phi_{j_1},\ldots,\phi_{j_t}\}$

Definition 9 *Let* $\langle\texttt{belfunction}\rangle\sigma_1,..,\sigma_n\langle\texttt{/belfunction}\rangle$ *be a BelVC or a BelSC, let* $S = \{(\psi_1,\kappa_1),$ $\ldots,(\psi_n,\kappa_n)\}$ *be the collection of (subset, mass) pairs in the component. This component adheres to the* **full belfunction distribution constraint** *iff the following two conditions hold:*

(1) $\Sigma_i\kappa_i = 1$
(2) *for all* $i,j,$ *if* $1 \le i \le n$ *and* $1 \le j \le n$ *and* $i \ne j,$ *then* $\psi_i \ne \psi_j$

When there are two BelVCs referring to the same textentry, we need to merge them. The following procedure implements Dempster's combination rule.

Definition 10 *([HL04a]) Let the following be two BelVCs*

$$\langle \texttt{belfunction} \rangle \sigma^1_1, .., \sigma^1_p \langle / \texttt{belfunction} \rangle$$
$$\langle \texttt{belfunction} \rangle \sigma^2_1, .., \sigma^2_q \langle / \texttt{belfunction} \rangle$$

where

1. *$\sigma^1_i \in \{\sigma^1_1, .., \sigma^1_p\}$ is of the form $\langle \texttt{mass value} = \kappa^1_i \rangle \psi^1_i \langle / \texttt{mass} \rangle$*
2. *the (subset, mass) pair collection is $S_1 = \{(\psi^1_1, \kappa^1_1), \ldots, (\psi^1_p, \kappa^1_p)\}$,*
3. *$\sigma^2_j \in \{\sigma^2_1, .., \sigma^2_q\}$ is of the form $\langle \texttt{mass value} = \kappa^2_j \rangle \psi^2_j \langle / \texttt{mass} \rangle$*
4. *the (subset, mass) pair collection is $S_2 = \{(\psi^2_1, \kappa^2_1), \ldots, (\psi^2_q, \kappa^2_q)\}$,*

Let the **combined BelVC** *be $\langle \texttt{belfunction} \rangle \sigma_1, .., \sigma_s \langle / \texttt{belfunction} \rangle$ where each $\sigma_k \in \{\sigma_1, .., \sigma_s\}$ is of the form $\langle \texttt{mass value} = \kappa_k \rangle \psi_k \langle / \texttt{mass} \rangle$ and*

$$\kappa_k = \frac{\Sigma \kappa^1_i \times \kappa^2_j}{1 - \Sigma \kappa^1_n \times \kappa^2_m}$$

such that $\psi_k = \psi^1_i \cap \psi^2_j$ for the (ψ^1_i, κ^1_i) and (ψ^2_j, κ^2_j) pairs, and $\psi^1_n \cap \psi^2_m = \emptyset$ for the (ψ^1_n, κ^1_n) and (ψ^2_m, κ^2_m) pairs, and ψ_k is of the form $\langle \texttt{massitem} \rangle \phi_{k_1} \langle / \texttt{massitem} \rangle, \ldots, \langle \texttt{massitem} \rangle \phi_{k_z} \langle / \texttt{massitem} \rangle$.

The value $\kappa_\perp = \Sigma \kappa^1_n \times \kappa^2_m$ (that is, $\Sigma_{A \cap B = \emptyset}\ (m_1(A) \times m_2(B))$ indicates how much of the total belief has been committed to the empty set while combining two pieces of uncertain information. A higher $\kappa_\perp$ value reflects either an inconsistency among the two sources or lower confidence in any of the possible outcomes from both sources.

Definition 11 *Let the abstract terms τ_1 and τ_2 each denote a BelVC and let X be a logical variable. The predicate $\texttt{Dempster}(\tau_1, \tau_2, X)$ is such that X is evaluated to τ_3 where τ_3 is the abstract term denoting the combined BelVC obtained by Definition 10.*

The predicate $\texttt{Dempster}(\tau_1, \tau_2, X)$ is defined in Prolog to carry out the actual merge. Looking back at Example 1 again, if we let τ_1 and τ_2 be the abstract terms for the first two XML documents in the example, then X represents the merged abstract term isomorphic to the third XML document in the example.

3 Merging uncertainty on textentries with compatible frames

In this section, we concentrate on merging structured reports with uncertain information (uncertainty valid components) on textentries where either the uncertainty is expressed at different levels of granularity (which we describe as *semantically homogeneous*) or on different but inter-related sets of values (which we describe as *semantically heterogeneous*). We consider both probabilistic and belief function information and take probability theory as a special case of belief function theory. We leave the topic of merging semantically heterogeneous uncertainty-valid components on subtrees from multiple structured reports to the next section.

When merging two structured reports, one with an uncertainty valid component and one without, we take the latter as a special case of the former and assign value 1.0 (no matter whether it stands for a probability value or a mass value) to the corresponding textentry (or textentries). Then, these two structured reports can be merged using one of the rules defined below.

Before proceeding to the details of this logic-based merging technique, we need to emphasize that in this paper any two uncertainty components to be merged are assumed to refer to the same or related issue (or topic) that are being considered. For instance, both uncertainty components are either about the `deposit` of `layer X` of `NorthSea` for `WellNo A`, or about the `deposit` or `lithology` of `NorthSea` for `WellNo A`, `layer Y`. If it is the case that one uncertainty component is about the `deposit` of `NorthSea` for `WellNo A` and another is about the `lithology` of `NorthSea` for `WellNo B`, then these two uncertainty components cannot be merged. The method to verify semantically whether two given uncertainty components are eligible for merging is given in [HS04]. In the rest of this paper, whenever we intend to merge two such components, we assume their eligibility has been checked and we will not repeat this prerequisite any further.

3.1 Propagation operation in DS theory

When two mass functions are not given on the same frame, they cannot be combined directly, rather one mass function has to be propagated to the frame of another mass function. Let us now look at several situations when this propagation can take place.

Definition 12 *Let Ω_1 and Ω_2 be two frames of discernment and Γ be a mapping function $\Gamma : \Omega_1 \rightarrow 2^{\Omega_2}$. When the following conditions hold, Ω_2 is called a* **refinement** *of Ω_1, and Ω_1 is called a* **coarsening** *of Ω_2. Γ is called a* **refinement mapping**.

$$\begin{array}{ll} (1)\ \Gamma(\phi) = T_\phi \neq \emptyset, & \textit{for all } \phi \in \Omega_1, \ \ \textit{where } T_\phi \subseteq \Omega_2 \\ (2)\ \Gamma(\phi_i) \cap \Gamma(\phi_j) = \emptyset, & \textit{when } \ i \neq j \\ (3)\ \cup_{\phi \in \Omega_1} \Gamma(\phi) = \Omega_2 & \end{array}$$

Example 2 in Section 1 gives a mass function (we take a probability distribution as a special case of mass function) on frame $\Omega_1 = \{\texttt{liquid}, \texttt{solid}\}$ and another on frame $\Omega_2 = \{\texttt{water}, \texttt{oil}, \texttt{gas}, \texttt{sand}, \texttt{stone}\}$ respectively. Ω_2 is in fact a refinement of Ω_1, if we define the refinement mapping function Γ as

$$\Gamma(\texttt{liquid}) = \{\texttt{water}, \texttt{oil}, \texttt{gas}\}, \ \ \Gamma(\texttt{solid}) = \{\texttt{sand}, \texttt{stone}\}.$$

A refinement mapping generates a set of disjoint subsets of the finer frame. Through a refinement mapping Γ, we can also define a **coarsening mapping** function $\Gamma' : \Omega_2 \rightarrow \Omega_1$ as:

$$\Gamma'(\psi) = \phi \quad \text{where} \quad \psi \in T_\phi \quad \text{and } \Gamma(\phi) = T_\phi$$

For instance, the coarsening mapping function of the above refinement mapping function gives

$$\Gamma'(\texttt{water}) = \Gamma'(\texttt{oil}) = \Gamma'(\texttt{gas}) = \texttt{liquid} \ \ \Gamma'(\texttt{sand}) = \Gamma'(\texttt{stone}) = \texttt{solid}$$

Lemma 1 *Let Ω_2 be a refinement of frame Ω_1 by mapping function Γ and let m_{Ω_1} be a mass function on Ω_1. Function m_{Ω_2} defined below is a mass function on Ω_2.*

$$m_{\Omega_2}(T) = m_{\Omega_1}(S) \ \ \textit{where } T = \bigcup \Gamma(\phi) \ \ \textit{for } \phi \in S, \textit{and } S \subseteq \Omega_1 \textit{ is a focal element}. \qquad (1)$$

Let Ω_1 and Ω_2 be two frames as defined in Example 2 and let

$$m_{\Omega_1}(\{\texttt{liquid}\}) = 0.4, \;\; m_{\Omega_1}(\{\texttt{solid}\}) = 0.6$$

be a mass function on Ω_1. Applying Lemma 1,

$$m_{\Omega_2}(\{\texttt{water}, \texttt{oil}, \texttt{gas}\}) = 0.4 \;\; m_{\Omega_2}(\{\texttt{sand}, \texttt{stone}\}) = 0.6$$

is a mass function on Ω_2.

Lemma 2 *Let Ω_1 be a coarsening of frame Ω_2 by coarsening mapping function Γ' and let m_{Ω_2} be a mass function on Ω_2. Function m_{Ω_1} defined below is a mass function on Ω_1.*

$$m_{\Omega_1}(S) = \Sigma_T \; m_{\Omega_2}(T) \;\; \textit{where } S = \bigcup \Gamma'(\psi) \textit{ for } \; \psi \in T \textit{ and } T \subseteq \Omega_2 \textit{ is a focal element.} \tag{2}$$

Yet again, if we have $m_{\Omega_2}(\{\texttt{water}, \texttt{oil}\}) = 0.2$ and $m_{\Omega_2}(\{\texttt{gas}\}) = 0.8$, based on Lemma 2, this mass function generates a mass function on Ω_1 as $m_{\Omega_1}(\{\texttt{liquid}\}) = 0.2 + 0.8 = 1$.

Now we look at more complex mapping relations between frames.

Definition 13 *Let Ω_1 and Ω_2 be two frames of discernment containing possible values to two related questions Q_1 and Q_2. Let Γ be a mapping function $\Gamma : \Omega_1 \to 2^{\Omega_2}$ which defines that whenever ϕ^1_i is the true answer to question Q_1 then the true answer to question Q_2 must be one of the elements in $\Gamma(\phi^1_i) \neq \emptyset$, and for every $\phi^2_j \in \Omega_2$, there exists at least one ϕ^1_i such that $\phi^2_j \in \Gamma(\phi^1_i)$. Then frames Ω_1 and Ω_2 are said to be* **compatible***.*

Mapping Γ is referred to as a **compatibility mapping** [LGS86, LH+93]. Equally, a compatibility mapping can be defined from Ω_2 to Ω_1. A refinement (or coarsening) mapping is a special case of compatibility mapping.

Lemma 3 *Let Ω_1 and Ω_2 be two related frames with a compatibility mapping Γ. Let m_{Ω_1} be a mass function on Ω_1. Then function m_{Ω_2} defined below is a mass function on Ω_2.*

$$m_{\Omega_2}(T) = \Sigma_S \; m_{\Omega_1}(S) \textit{ where } T = \bigcup \Gamma(\phi) \textit{ for } \phi \in S \textit{ and } S \subseteq \Omega_1 \textit{ is a focal element.} \tag{3}$$

All these three Lemmas can be proved easily (e.g., [Sha76]).

For instance, the relationship between *deposits* (captured by Ω_2) and *lithologies* (captured by Ω_3) can be established through a mapping $\Gamma : \Omega_2 \to 2^{\Omega_3}$ as

$$\begin{array}{lll} \Gamma(\texttt{water}) = \{\texttt{L1}, \texttt{L2}\}, & \Gamma(\texttt{oil}) = \{\texttt{L3}, \texttt{L4}\}, & \Gamma(\texttt{gas}) = \{\texttt{L2}, \texttt{L5}, \texttt{L6}\}, \\ \Gamma(\texttt{sand}) = \{\texttt{L8}, \texttt{L9}\}, & \Gamma(\texttt{stone}) = \{\texttt{L7}, \texttt{L8}\}. & \end{array}$$

Or a mapping function $\Gamma'' : \Omega_3 \to 2^{\Omega_2}$ as

$$\begin{array}{lll} \Gamma''(\texttt{L1}) = \{\texttt{water}\}, & \Gamma''(\texttt{L2}) = \{\texttt{water}, \texttt{gas}\}, & \Gamma''(\texttt{L3}) = \{\texttt{oil}\}, \\ \Gamma''(\texttt{L4}) = \{\texttt{oil}\}, & \Gamma''(\texttt{L5}) = \{\texttt{gas}\}, & \Gamma''(\texttt{L6}) = \{\texttt{gas}\}, \\ \Gamma''(\texttt{L7}) = \{\texttt{stone}\}, & \Gamma''(\texttt{L8}) = \{\texttt{sand}, \texttt{stone}\}, & \Gamma''(\texttt{L9}) = \{\texttt{sand}\}. \end{array}$$

Using this mapping relationship, the uncertain information on Ω_3 in the second XML document in Example 3 can be propagated to Ω_2 to obtain a new mass function on `deposit` as

$$m_{\Omega_3}(\{\texttt{water}, \texttt{oil}\}) = 0.3, \;\; m_{\Omega_3}(\{\texttt{water}, \texttt{gas}\}) = 0.7.$$

3.2 Predicate for belief propagation on textentries

We now define a formal procedure to perform the above propagations as discussed in Section 3.1 and define a predicate to call the procedure.

Definition 14 *Let $\langle\texttt{belfunction}\rangle\sigma_1^1, .., \sigma_p^1\langle/\texttt{belfunction}\rangle$ be a BelVC where*

1. *$\sigma_i^1 \in \{\sigma_1^1, .., \sigma_p^1\}$ is of the form $\langle\texttt{mass value} = \kappa_i^1\rangle\psi_i^1\langle/\texttt{mass}\rangle$*
2. *$S = \{(\psi_1^1, \kappa_1^1), \ldots, (\psi_p^1, \kappa_p^1)\}$ is the collection of (subset, mass) pairs*
3. *$\Gamma : \Omega_1 \rightarrow 2^{\Omega_2}$ is a compatibility mapping and $\Gamma(\psi_i^1) = \Gamma(\phi_{i_1}^1) \cup \ldots \cup \Gamma(\phi_{i_x}^1)$ where $\psi_i^1 = \{\phi_{i_1}^1, \ldots, \phi_{i_x}^1\}$*

Let the propagated BelVC on Ω_2 be $\langle\texttt{belfunction}\rangle\sigma_1^2, .., \sigma_q^2\langle/\texttt{belfunction}\rangle$ where each $\sigma_j^2 \in \{\sigma_1^2, .., \sigma_q^2\}$ is of the form $\langle\texttt{mass value} = \kappa_j^2\rangle\psi_j^2\langle/\texttt{mass}\rangle$ and

$$\kappa_j^2 = \Sigma_i \kappa_i^1 \;\; \textit{s.t} \;\; \psi_j^2 = \Gamma(\psi_i^1) \;\; \textit{for each} \;\; (\psi_i^1, \kappa_i^1) \; \textit{pair}$$

and ψ_j^2 is of the form $\langle\texttt{massitem}\rangle\phi_{j_1}^2\langle/\texttt{massitem}\rangle \cdots \langle\texttt{massitem}\rangle\phi_{j_y}^2\langle/\texttt{massitem}\rangle$

Definition 15 *Let the abstract term τ be a BelVC on Ω_1. Let Γ be a compatibility mapping $\Gamma : \Omega_1 \rightarrow 2^{\Omega_2}$, and X be a logical variable. The predicate $\texttt{Propagate}(\tau, \Gamma, X)$ is such that X is evaluated to τ' where τ' is the abstract term denoting the propagated BelVC on Ω_2 obtained by Definition 14.*

Predicate $\texttt{Propagate}(\tau, \Gamma, X)$ can be used to generate a BelVC on a frame from an existing BelVC on another frame, no matter whether the relationship between the two frames is a refinement, or a coarsening, or compatible.

Since we take a ProVC as a special case of BelVCs, it is possible to easily convert the former to the format of the latter as given in [HL04a]. We repeat this definition again here.

Definition 16 *Let abstract term τ be a ProVC $\langle\texttt{probability}\rangle\sigma_1, .., \sigma_n\langle/\texttt{probability}\rangle$ and each $\sigma_i \in \{\sigma_1, .., \sigma_n\}$ is of the form $\langle\texttt{prob value} = \kappa\rangle\phi\langle/\texttt{prob}\rangle$ where $\kappa \in [0, 1]$ and ϕ is a textentry. Then τ' is the abstract term denoting the BelVC $\langle\texttt{belfunction}\rangle\sigma'_1, .., \sigma'_{\mathrm{n}}\langle/\texttt{belfunction}\rangle$ where each $\sigma'_i \in \{\sigma'_1, .., \sigma'_n\}$ is of the form $\langle\texttt{mass value} = \kappa\rangle\langle\texttt{massitem}\rangle\phi\langle/\texttt{massitem}\rangle\langle/\texttt{mass}\rangle$ and $\kappa \in [0, 1]$, and ϕ is a textentry.*

Definition 17 *If the abstract term τ is a ProVC and X is a logical variable, then $\texttt{BayesBelief}(\tau, X)$ is a predicate such that X is evaluated to τ' where τ' is the abstract term denoting the BelVC obtained from τ by Definition 16.*

In an analogous way to Definitions 16 and 17, it is possible to define how a ProSC can be converted into a BelSC.

Definition 18 *Let abstract term τ be a ProSC $\langle\texttt{probability}\rangle\sigma_1, .., \sigma_n\langle/\texttt{probability}\rangle$ and each $\sigma_i \in \{\sigma_1, .., \sigma_n\}$ is of the form $\langle\texttt{prob value} = \kappa_i\rangle\psi_i\langle/\texttt{prob}\rangle$ where $\kappa_i \in [0, 1]$ and ψ_i is in the form*

$$\langle\texttt{multiitem}\rangle\langle\rho_{i_1}\rangle\phi_{i_1}\langle/\rho_{i_1}\rangle \ldots \langle\rho_{i_x}\rangle\phi_{i_x}\langle/\rho_{i_x}\rangle\langle/\texttt{multiitem}\rangle$$

Then τ' is the abstract term denoting the BelVC $\langle\texttt{belfunction}\rangle\sigma'_1, .., \sigma'_n\langle/\texttt{belfunction}\rangle$ where each $\sigma'_i \in \{\sigma'_1, .., \sigma'_n\}$ is of the form $\langle\texttt{mass value} = \kappa_{\mathtt{i}}\rangle\psi'_{\mathtt{i}}\langle/\texttt{mass}\rangle$ and $\kappa_i \in [0, 1]$, and ψ'_i is in the form

$$\langle\texttt{multiitem}\rangle\langle\rho_{i_1}\rangle\phi_{i_1}\langle/\rho_{i_1}\rangle \ldots \langle\rho_{i_x}\rangle\phi_{i_x}\langle/\rho_{i_x}\rangle\langle/\texttt{multiitem}\rangle$$

Definition 19 *Let the abstract term τ be a ProSC and let X be a logical variable. The predicate* $\texttt{BayesBelief}(\tau, X)$ *is such that X is evaluated to τ' where τ' is the abstract term denoting the BelSC obtained from τ by Definition 18.*

Example 10 *Let us re-visit Example 2. Let τ_1 and τ_2 be the abstract terms for the two XML documents in this example left and right. Both of the ProVCs can be converted by calling predicates* $\texttt{BayesBelief}(\tau_1, X_1)$ *and* $\texttt{BayesBelief}(\tau_2, X_2)$*, where X_1 and X_2 are ground by abstract terms τ'_1 and τ'_2 respectively where τ'_1 and τ'_2 are the converted BelVCs represented by the XML documents left and right below (respectively).*

```
⟨report⟩                            ⟨report⟩
 ⟨deposit⟩                           ⟨deposit⟩
  ⟨belfunction⟩                       ⟨belfunction⟩
   ⟨mass value = “0.2”                 ⟨mass value = “0.4”
    ⟨massitem⟩water⟨/massitem⟩          ⟨massitem⟩liquid⟨/massitem⟩
   ⟨/mass⟩                             ⟨/mass⟩
   ⟨mass value = “0.8”⟩                ⟨mass value = “0.6”⟩
    ⟨massitem⟩sand⟨/massitem⟩           ⟨massitem⟩solid⟨/massitem⟩
   ⟨/mass⟩                             ⟨/mass⟩
  ⟨/belfunction⟩                      ⟨/belfunction⟩
 ⟨/deposit⟩                          ⟨/deposit⟩
⟨/report⟩                           ⟨/report⟩
```

If an agent's query is posed on the concept `deposit` *at the general level, e.g, either answer* `solid` *or* `liquid` *will be sufficient, then uncertain information represented by X_1 should be propagated to this general frame using predicate* $\texttt{Propagate}(X_1, \Gamma', X_3)$ *where Γ' is a coarsening mapping and X_3 is ground to τ_3 as follows.*

```
⟨belfunction⟩
 ⟨mass value = “0.2”⟩
  ⟨massitem⟩liquid⟨/massitem⟩
 ⟨/mass⟩
 ⟨mass value = “0.8”⟩
  ⟨massitem⟩solid⟨/massitem⟩
 ⟨/mass⟩
⟨/belfunction⟩
```

Finally, τ_3 can be combined with τ'_2 using $\texttt{Dempster}(X_3, X_2, X_4)$ *to obtain the final result where X_3 is ground by τ_3 and X_2 is ground by τ'_2. The whole sequence of calls to the Prolog predicates can be summarized as:*

$$\texttt{BayesBelief}(\tau_1, X_1) \wedge \texttt{BayesBelief}(\tau_2, X_2) \wedge \texttt{Propagate}(X_1, \Gamma', X_3) \wedge \texttt{Dempster}(X_3, X_2, X_4)$$

On the other hand, if a query is posed at a more detailed level, then the call to $\texttt{Propagate}(X_1, \Gamma', X_3)$ *is replaced by* $\texttt{Propagate}(X_2, \Gamma, X_3)$ *where the mass function on the general level of frame will be propagated to the finer frame through a refinement mapping Γ. In this case, the sequence of executions of predicates is revised as:*

$$\texttt{BayesBelief}(\tau_1, X_1) \wedge \texttt{BayesBelief}(\tau_2, X_2) \wedge \texttt{Propagate}(X_2, \Gamma, X_3) \wedge \texttt{Dempster}(X_1, X_3, X_4)$$

Example 11 *Consider the following three uncertainty valid components where τ_1, τ_2 are the abstract* *terms of the left and right BelVCs, and τ_3 is the corresponding abstract term for the ProVC.*

```
⟨belfunction⟩                        ⟨belfunction⟩
 ⟨mass value = “0.2”⟩                 ⟨mass value = “0.4”⟩
  ⟨massitem⟩water⟨/massitem⟩            ⟨massitem⟩liquid⟨/massitem⟩
  ⟨massitem⟩gas⟨/massitem⟩
 ⟨/mass⟩                              ⟨/mass⟩
 ⟨mass value = “0.8”⟩                 ⟨mass value = “0.6”⟩
  ⟨massitem⟩sand⟨/massitem⟩             ⟨massitem⟩solid⟨/massitem⟩
 ⟨/mass⟩                              ⟨/mass⟩
⟨/belfunction⟩                       ⟨/belfunction⟩
```

```
⟨probability⟩
 ⟨prob value = “0.2”⟩water⟨/prob⟩
 ⟨prob value = “0.8”⟩gas⟨/prob⟩
⟨/probability⟩
```

Let Q be an agent's query about information on the possible deposit of a certain location at the most general level. Q can be answered by the following string of calls to several predicates.

$$\begin{aligned}&\texttt{Propagate}(\tau_1, \Gamma', X_1) \wedge \texttt{Dempster}(X_1, \tau_2, X_2)\\ &\qquad \wedge\texttt{BayesBelief}(\tau_3, X_3) \wedge \texttt{Propagate}(X_3, \Gamma', X_4) \wedge \texttt{Dempster}(X_2, X_4, X_5)\end{aligned}$$

In this, $\texttt{Propagate}(\tau_1, \Gamma', X_1) \wedge \texttt{Dempster}(X_1, \tau_2, X_2)$ *takes the converted BelVC (from a detailed frame to a general frame through a coarsening mapping Γ') as its first argument and combines it with the second BelVC (on the right-hand side) to produce a merged result denoted by X_2. This newly generated BelVC is then combined with another converted BelVC denoted by variable X_4 (from probability, using condition literal* $\texttt{BayesBelief}(\tau_3, X_3)$ *first and then propagated to the right frame) using predicate* `Dempster` *to obtain the final result which is denoted by X_5.*

```
⟨belfunction⟩
 ⟨mass value = “0.039”⟩
  ⟨massitem⟩liquid⟨/massitem⟩
 ⟨/mass⟩
 ⟨mass value = “0.960”⟩
  ⟨massitem⟩solid⟨/massitem⟩
 ⟨/mass⟩
⟨/belfunction⟩
```

All the three sources have a high confidence in choice $\{\texttt{solid}\}$ than in $\{\texttt{liquid}\}$, so the combined result gives a higher confidence in the choice preferred by all of them and a lower confidence in the less preferred one. This is due to the fact that these sources are in agreement with each other. Therefore, when multiple sources are not in conflict, merging them will produce a more complete and comprehensive solution than individual sources. Methods for detecting inconsistencies among multiple sources have been discussed and provided in [HL04a] and we will not discuss them further here.

4 Merging uncertain information on subtrees

To develop predicates for merging subtree uncertainty components, we need to look at the approaches to propagating mass functions among compound frames, since a subtree uncertainty component contains two or more frames of discernment whilst a query may only be related to one of them. The first subsection below looks into the techniques of mass function propagation in this situation which is followed by a subsection on predicates to merge subtree uncertainty components.

4.1 Extension and projection operations in DS theory

The concept of compatible frames (or compatibility relations) can be extended to situations where a frame is in fact a Cartesian product (or a subset of the product) of several frames.

Definition 20 *Let Ω_i, $i = 1, \dots, n$ be n frames of discernment each of which contains mutually exclusive and exhaustive solutions to a related question or a variable. Frame $\Omega = \otimes_i \Omega_i$ is a* **joint frame** *containing solutions to the joint question or the joint variable.*

For instance, let Ω_1 be a frame containing values (answers) to question $'$`What deposit is it?`$'$, and Ω_2 be a frame containing values (answers) to question $'$`What lithology is it?`$'$, then $\Omega_1 \otimes \Omega_2$ is the frame containing values for the joint question $'$`What deposit and what lithology is it?`$'$ with values that are in the form $< \omega_{1_i}, \omega_{2_j} >$ where $\omega_{1_i} \in \Omega_1$ and $\omega_{2_j} \in \Omega_2$. If some of the pairs $< \omega_{1_i}, \omega_{2_j} >$ are false, that is, ω_{1_i} and ω_{2_j} are not compatible, $\Omega_1 \otimes \Omega_2$ is then a proper subset of the set product consisting of only those pairs with compatible elements from individual frames. Values of a frame are also referred to as **configurations** of the question or variable associated with the frame. For example, if we let g and h be two variables that can take values from Ω_1 and $\Omega_1 \otimes \Omega_2$ respectively, then value `water` is a configuration of g and value $< \texttt{water}, \texttt{L1} >$ is a configuration of h.

Definition 21 *[LHA03] Let $V = \{r_1, r_2, \dots, r_n\}$ be n variables each of which has a set of configurations represented by its associated frame of discernment Ω_i. Let $V_p \subseteq V$ and $V_q \subseteq V$ be two subsets of variables where $V_p \subset V_q$, and let $\Omega_{V_p} = \otimes_{r_i \in V_p} \Omega_i$ and $\Omega_{V_q} = \otimes_{r_j \in V_q} \Omega_j$ be two joint frames for them. Let $Q \subseteq \Omega_{V_q}$ be a set of configurations of V_q. Then, the* **projection** *of Q to Ω_{V_p}, denoted by $Q^{\downarrow V_p}$ is a set of configurations for V_p. Similarly, let H be a subset of Ω_{V_p}, then the* **extension** *of H to Ω_{V_q}, denoted by $H^{\uparrow V_q}$ is $H \otimes \Omega_{V_q \setminus V_p}$ which is a set of configurations for variable set V_q.*

V_p, a subset of variables, is also referred to as a joint variable. In the following, we talk about a subset of variables or a joint variable interchangeably without further explanation. In either case, Ω_{V_p} is the full set of configurations for it.

Example 12 *Assume r_1, r_2, r_3, and r_4 are four variables taking values from frames of discernment Ω_i, $i = 1, 2, 3, 4$ respectively, where $\Omega_1 = \{\omega_{11}, \omega_{12}\}$, $\Omega_2 = \{\omega_{21}, \omega_{22}, \omega_{23}\}$, $\Omega_3 = \{\omega_{31}, \omega_{32}, \omega_{33}\}$, and $\Omega_4 = \{\omega_{41}, \omega_{42}, \omega_{43}, \omega_{44}\}$. Let $V_p = \{r_1, r_2\}$ and $V_q = \{r_1, r_2, r_3\}$ be two subsets of variables and $Q = \{< \omega_{11}, \omega_{21}, \omega_{31} >, < \omega_{12}, \omega_{23}, \omega_{31} >\}$ be a set of configurations for V_q, then $Q^{\downarrow V_p} = \{< \omega_{11}, \omega_{21} >, < \omega_{12}, \omega_{23} >\}$ is a set of configurations for V_p.*

However, given a set of configurations $H = \{< \omega_{11}, \omega_{21} >, < \omega_{12}, \omega_{23} >\}$ for V_p, its extension to variable set V_q would be $Q' = H^{\uparrow V_q} = \{< \omega_{11}, \omega_{21}, \omega_{31} >, < \omega_{12}, \omega_{23}, \omega_{31} >, < \omega_{11}, \omega_{21}, \omega_{32} >, < \omega_{12}, \omega_{23}, \omega_{32} >, < \omega_{11}, \omega_{21}, \omega_{33} >, < \omega_{12}, \omega_{23}, \omega_{33} >\}$. This set of configurations is different from Q although the projection of Q is H too.

Definition 22 *Let $V_p \subseteq V$ and $V_q \subseteq V$ be two subsets of variables where $\emptyset \neq V_p \subset V_q$. Let m be a mass function on Ω_{V_q} for the joint variable V_q, then the* **marginal** *of m on Ω_{V_p} for the joint variable V_p, denoted by $m^{\downarrow V_p}$ defined below, is a mass function on Ω_{V_p}*

$$m^{\downarrow V_p}(H) = \Sigma\{m(G) | G \subseteq \Omega_{V_q}, G^{\downarrow V_p} = H,\ G \textit{ is a focal element}\}$$

Equally, if m is a mass function on Ω_{V_p} for the joint variable V_p, then the **marginal** *of m on Ω_{V_q} for the joint variable V_q, denoted by $m^{\uparrow V_q}$ defined below, is a mass function on Ω_{V_q}*

$$m^{\uparrow V_q}(G) = \Sigma\{m(H) | H \subseteq \Omega_{V_p}, H^{\uparrow V_q} = G,\ H \textit{ is a focal element}\}$$

Example 13 *Consider the following two uncertainty components.*

```
⟨belfunction⟩
  ⟨mass value = “0.2”⟩
    ⟨massitem⟩water⟨/massitem⟩
    ⟨massitem⟩gas⟨/massitem⟩
  ⟨/mass⟩
  ⟨mass value = “0.8”⟩
    ⟨massitem⟩sand⟨/massitem⟩
  ⟨/mass⟩
⟨/belfunction⟩
```

```
⟨belfunction⟩
  ⟨mass value = “0.4”⟩
    ⟨multiitem⟩
      ⟨deposit⟩water⟨/deposit⟩
      ⟨lithology⟩L1⟨/lithology⟩
    ⟨/multiitem⟩
    ⟨multiitem⟩
      ⟨deposit⟩oil⟨/deposit⟩
      ⟨lithology⟩L3⟨/lithology⟩
    ⟨/multiitem⟩
  ⟨/mass⟩
  ⟨mass value = “0.6”⟩
    ⟨multiitem⟩
      ⟨deposit⟩gas⟨/deposit⟩
      ⟨lithology⟩L2⟨/lithology⟩
    ⟨/multiitem⟩
  ⟨/mass⟩
⟨/belfunction⟩
```

The left-hand XML document defines a mass function on frame $\Omega_2 = \{\texttt{water}, \texttt{oil}, \texttt{gas}, \texttt{sand}, \texttt{stone}\}$ as $m_{\Omega_2}(\{\texttt{water}, \texttt{gas}\}) = 0.2$ and $m_{\Omega_2}(\{\texttt{sand}\}) = 0.8$, and the right-hand XML document defines another mass function on frame $\Omega_2 \otimes \Omega_3$ where $\Omega_3 = \{\texttt{L1}, \texttt{L2}, \texttt{L3}, \texttt{L4}, \texttt{L5}\}$ as $m_{\Omega_2 \otimes \Omega_3}(\{< \texttt{water}, \texttt{L1} >, < \texttt{oil}, \texttt{L3} >\}) = 0.4$ and $m_{\Omega_2 \otimes \Omega_3}(\{< \texttt{gas}, \texttt{L2} >\}) = 0.6$. Assume an agent is interested in knowing the joint impact of these two pieces of evidence on the value set Ω_2, then the impact of the mass function on $\Omega_2 \otimes \Omega_3$ has to be marginalized on Ω_2. Based on Definition 22, $m_{\Omega_2 \otimes \Omega_3}$ gives a new mass function on Ω_2 as $m'_{\Omega_2}(\{\texttt{water}, \texttt{oil}\}) = 0.4$ and $m'_{\Omega_2}(\{\texttt{gas}\}) = 0.6$, which can be merged with m_{Ω_2} using $\texttt{Dempster}(\tau_1, \tau_2, X)$ to obtain the final result $m(\{\texttt{water}\}) = 0.4$ and $m(\{\texttt{gas}\}) = 0.6$, if we assume that these two pieces of evidence are from independent sources.

4.2 Predicate for belief marginalization on subtrees

Now we provide a procedure that implements the marginalization of a mass function from a larger variable set to a smaller set defined in Definition 22.

Definition 23 *Let the following be a BelSC $\langle\texttt{belfunction}\rangle\sigma_1^1, .., \sigma_q^1\langle\texttt{/belfunction}\rangle$ where*

1. *$\sigma_i^1 \in \{\sigma_1^1, .., \sigma_q^1\}$ is of the form $\langle\texttt{mass value} = \kappa_i^1\rangle\ \psi_i^1\ \langle\texttt{/mass}\rangle$*
2. *ψ_i^1 is of the form $\langle\texttt{multiitem}\rangle\varphi_{i_1}^1\langle\texttt{/multiitem}\rangle \ldots \langle\texttt{multiitem}\rangle\varphi_{i_n}^1\langle\texttt{/multiitem}\rangle$*
3. *each $\varphi_{i_t}^1$ in $\{\varphi_{i_1}^1, \ldots, \varphi_{i_n}^1\}$ is of the form $\langle\rho_{i_{t1}}^1\rangle\phi_{i_{t1}}^1\langle/\rho_{i_{t1}}^1\rangle, \ldots, \langle\rho_{i_{tl}}^1\rangle\phi_{i_{tl}}^1\langle/\rho_{i_{tl}}^1\rangle$*
4. *and $\rho_{i_{t1}}^1, \ldots, \rho_{i_{tl}}^1$ are tagnames, and $\phi_{i_{t1}}^1, \ldots, \phi_{i_{tl}}^1$ are textentries.*

Let the variable set associated with it be V_q with configurations in Ω_{V_q}. Let $V_p \subset V_q$.

When $|V_p| > 1$, let the marginalized BelSC on Ω_{V_p} be

$$\langle\texttt{belfunction}\rangle\sigma_1^2, .., \sigma_p^2\langle\texttt{/belfunction}\rangle$$

where each $\sigma_j^2 \in \{\sigma_1^2, .., \sigma_p^2\}$ is of the form $\langle\texttt{mass value} = \kappa_j^2\rangle\psi_j^2\langle\texttt{/mass}\rangle$

and each ψ_j^2 is of the form

$$\langle\texttt{multiitem}\rangle\varphi^2_{j_1}\langle/\texttt{multiitem}\rangle \ldots \langle\texttt{multiitem}\rangle\varphi^2_{j_m}\langle/\texttt{multiitem}\rangle$$

and each $\varphi^2_{j_k}$ is of the form

$$\langle\rho^2_{j_{k1}}\rangle\phi^2_{j_{k1}}\langle/\rho^2_{j_{k1}}\rangle, \ldots, \langle\rho^2_{j_{kf}}\rangle\phi^2_{j_{kf}}\langle/\rho^2_{j_{kf}}\rangle$$

and

$$\kappa^2_j = \Sigma_i\kappa^1_i, \textit{s.t.}, \{< \phi^1_{i_{t1}}, \ldots, \phi^1_{i_{tl}} >\}^{\downarrow V_p} = \{< \phi^2_{j_{k1}}, \ldots, \phi^2_{i_{kf}} >\}$$

When $|V_p| = 1$, let the marginalized BelVC on Ω_{V_p} be

$$\langle\rho^2\rangle\langle\texttt{belfunction}\rangle\sigma^2_1, .., \sigma^2_p\langle/\texttt{belfunction}\rangle\langle/\rho^2\rangle$$

where each $\sigma^2_j \in \{\sigma^2_1, .., \sigma^2_p\}$ is of the form $\langle\texttt{mass value} = \kappa^2_j\rangle\psi^2_j\langle/\texttt{mass}\rangle$

and each ψ^2_j is of the form

$$\langle\texttt{massitem}\rangle\phi^2_{j_1}\langle/\texttt{massitem}\rangle, \ldots, \langle\texttt{massitem}\rangle\phi^2_{j_m}\langle/\texttt{massitem}\rangle$$

and

$$\kappa^2_j = \Sigma_i\kappa^1_i \textit{ such that }, \{< \phi^1_{i_{t1}}, \ldots, \phi^1_{i_{tl}} >\}^{\downarrow V_p} = \{\phi^2_{j_z}\} \textit{ and } \phi^2_{j_z} \in \{\phi^2_{j_1}, \ldots, \phi^2_{j_m}\}$$

and ρ^2 is a tagname that is associated with the set of values in Ω_{V_p}.

When $| \ V_p \ |= 1$, that is, there is only one variable in set V_p, a BelSC is reduced to a BelVC.

Definition 24 *Let the abstract term τ be a BelSC on a subtree with variable set V_q. Let V_p be a subset of V_q and X be a logical variable. The predicate* $\texttt{PropagateTree}(\tau, V_q, X)$ *is such that X is evaluated to τ' where τ' is the abstract term denoting the propagated BelSC (or BelVC) on V_p obtained by Definition 23.*

Example 14 *Let τ denote the BelSC in Example 13. Applying predicate* $\texttt{PropagateTree}(\tau, \{\texttt{deposit}\}, X)$, *we obtain a new BelVC as*

```
⟨deposit⟩
 ⟨belfunction⟩
  ⟨mass value = “0.4”⟩
   ⟨massitem⟩water⟨/massitem⟩
   ⟨massitem⟩oil⟨/massitem⟩
  ⟨/mass⟩
  ⟨mass value = “0.6”⟩
   ⟨massitem⟩gas⟨/massitem⟩
  ⟨/mass⟩
 ⟨/belfunction⟩
⟨/deposit⟩
```

Since there is only one variable to project on when using this predicate, a subtree structure is reduced to a BelVC on a textentry.

Similar to the procedure and predicate above, it is possible to define another procedure and predicate to marginalize a mass function from Ω_{V_p} to Ω_{V_q} through an extension operation. However, obtaining a mass function on a larger frame (with more variables) is not as useful as the projection operation which derives a mass function on a smaller frame, therefore we will not include these detailed definitions in this paper.

5 Comparison with related approaches

In [NJ02], a probabilistic XML model was presented to deal with information with uncertainty that was in the form of probabilities. Using this model, we can construct an XML report as illustrated below. Two types of probability assignments are distinguished, mutually exclusive or not mutually exclusive. For the first type, probabilities are assigned to single atoms where only one of these atoms can be true, and the total sum of probability values is less than or equal to 1 (as for ⟨precipitation⟩). For the second type, two single atoms can be compatible, so the total sum of probabilities can be greater than 1 (as for ⟨cities⟩).

```
⟨report⟩
 ⟨source⟩TV1⟨/source⟩
 ⟨date⟩19/3/02⟨/date⟩
 ⟨cities⟩
  ⟨city Prob = “0.7”⟩
      ⟨cityName⟩London⟨/cityName⟩
      ⟨precipitation⟩
       ⟨Dist type = “mutually − exclusive”⟩
        ⟨Val Prob = “0.1”⟩sunny⟨/Val⟩
        ⟨Val Proc = “0.7”⟩rain⟨/Val⟩
       ⟨/Dist⟩
      ⟨/precipitation⟩
  ⟨/city⟩
  ⟨city Prob = “0.4”⟩
      ⟨cityName⟩GreaterLondon⟨/cityName⟩
      ⟨precipitation⟩
       ⟨Dist type = “mutually − exclusive”⟩
        ⟨Val Prob = “0.2”⟩sunny⟨/Val⟩
        ⟨Val Proc = “0.6”⟩rain⟨/Val⟩
       ⟨/Dist⟩
      ⟨/precipitation⟩
  ⟨/city⟩
 ⟨/cities⟩
⟨/report⟩
```

This model allows probabilities to be assigned to multiple granularities. When this occurs, the probability of an element is true is conditioned upon the existence of its parent (with probability), and so on until up to the root of the tree. For example, if we would like to know the probability of sunny in London, we have

$$\begin{aligned}
&\texttt{Prob}(\texttt{precipitation} = \texttt{sunny} \wedge \texttt{cityName} = \texttt{London})\\
&= \texttt{Prob}(\texttt{precipitation} = \texttt{sunny}) * \texttt{Prob}(\texttt{cityName} = \texttt{London})\\
&\qquad\qquad * \texttt{Prob}(\texttt{precipitation} = \texttt{sunny} \wedge \texttt{cityName} = \texttt{London} \mid \texttt{city}) * \texttt{Prob}(\texttt{city} \mid \texttt{cities})\\
&\qquad\qquad * \texttt{Prob}(\texttt{cities} \mid \texttt{report}) * \texttt{Prob}(\texttt{report})\\
&= 0.1 * 1.0 * 0.7 * 1.0 * 1.0 * 1.0 = 0.07
\end{aligned}$$

Therefore, the probability associated with a textentry (at any level) is treated as the conditional probability under its parent. A query is answered by tracing the relevant branches with the textentries specified by the query, and calculating probabilities using the conditional probabilities along these branches. These derived probabilities are then either multiplied or added depending on whether the “and” or the “or” operation are used in the original query. For instance, the query “London is either sunny or rain on 19/3/02” is evaluated

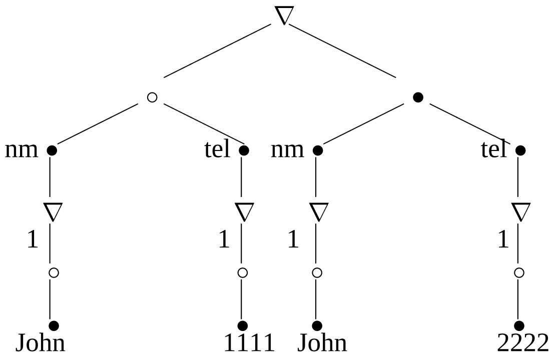

Figure 1: A probabilistic XML tree.

as:

$$\begin{aligned}
&\texttt{Prob}(\texttt{cityName} = \texttt{London} \wedge ((\texttt{precipitation} = \texttt{sunny}) \vee (\texttt{precipitation} = \texttt{rain})))\\
&= \texttt{Prob}(\texttt{cityName} = \texttt{London}) * \texttt{Prob}(\texttt{precipitation} = \texttt{sunny})\\
&\qquad *\texttt{Prob}(\texttt{cityName} = \texttt{London} \wedge \texttt{precipitation} = \texttt{sunny} \mid \texttt{city}) * \texttt{Prob}(\texttt{city} \mid \texttt{cities})\\
&\qquad *\texttt{Prob}(\texttt{cities} \mid \texttt{report}) * \texttt{Prob}(\texttt{report})\\
&\quad +\texttt{Prob}(\texttt{cityName} = \texttt{London}) * \texttt{Prob}(\texttt{precipitation} = \texttt{rain})\\
&\qquad *\texttt{Prob}(\texttt{cityName} = \texttt{London} \wedge \texttt{precipitation} = \texttt{rain} \mid \texttt{city}) * \texttt{Prob}(\texttt{city} \mid \texttt{cities})\\
&\qquad *\texttt{Prob}(\texttt{cities} \mid \texttt{report}) * \texttt{Prob}(\texttt{report})\\
&= (1.0 * 0.1 * 0.7 * *1.0 * 1.0 * 1.0) + (1.0 * 0.7 * 0.7 * 1.0 * 1.0 * 1.0) = 0.07 + 0.49 = 0.56.
\end{aligned}$$

The main advantage of this model is that it allows probabilities to be assigned to multiple levels of subtrees and provides a means to calculate the joint probability from them. However, it does not merge multiple probabilistic XML documents on the same issue. On the contrary, our uncertainty XML model focuses on multiple XML datasets and provides a set of means to merge opinions with uncertainty from different sources on textentries and subtrees. Therefore, our modelling and reasoning method is more general then that in [NJ02].

Another method to model and reason with probabilistic XML information is reported in [KKA05]. In this paper, three types of tags are identified as: (1) tags that stand for probabilities (denoted as ∇); (2) tags that stand for possible values associated with probabilities (denoted as ○); and (3) ordinary tag names (denoted as ●). A tree structure including these notations is illustrated in Figure 1 [KKA05].

Since the authors in the paper did not provide the actual XML structure for the example (or any other examples) to explicitly show how these types of tags are represented, we created an XML document for this example based on our own understanding as demonstrated in Figure 2 left. As we can see, there is lot of redundant information in this XML document, such as all the tags related to `possible values` are not strictly required, since a `possible` tag will always sit between a `probability` tag and a normal tag. This example can be equivalently represented in our ProSC format with a more compact structure as show in Figure 2 right.

Apart from the apparent structural differences between the approach in [KKA05] and ours, the real difference lies in the merging process itself. In [KKA05], each pair of (tag, value) and the combination of these pairs are treated as *possible worlds*. The merging of two probabilistic XML documents is to generate all the combinations of possible worlds from the two documents. As a consequence, there can be a huge number of branches in the merged XML document and there can be varieties of the document. For instance, one example given in the paper consists of two simple XML documents about `persons` with certainty (no probabilities). One document has details for four `persons` with each person has tags `firstname`, `lastname`, `phone`, `room` and associated values, the other document has details for two `persons` with the same set of tag names and corresponding values. Interestingly, merging these two simple documents in [KKA05] generates 3201 possible worlds which results in a very large and complex tree. Most of the branches in the tree are completely meaningless. However, using our logic-based merging tool, coupled with the background knowledge that only persons with the same firstname and lastname may refer to the same person, the merging result is a very simple XML document with four segments for four persons and with some probability components to indicate multiple values for same tags, such as `room`.

Issues of managing uncertain information on multiple levels of subtrees can be dealt with using *discounting* operation in DS theory and it will provide the same effect as the conditional probabilities in [NJ02]. Discussions on modelling and merging this type of uncertain information are considered in [HL04b].

For semantically heterogeneous uncertain information, we have mainly concentrated on semantic meanings of concepts that carry the uncertain information. In this paper we have not considered the semantic heterogeneity of XML branches. Also these issues have not been discussed in [NJ02] and [KKA05]. To illustrate a problem arising with semantic heterogeneity of XML branches, it is possible that we may wish to treat some branches as equivalent to others. So for example, in a particular application, we may wish to consider a branch `report/deposit/zone` and a branch `report/deposit/sector` as carrying information for the same location if they both have the same textentry. It is possible to axiomatize such equivalences on an application-dependent basis.

6 Conclusion

This paper significantly extends our previous paper [HL04a] on merging structured reports that contain uncertain information. In this paper, we have discussed methods to model and merge mass functions and probabilities assigned to textentries with different levels of granularity or to textentries which are interrelated, and to subtrees involving multiple frames where uncertain information is semantic heterogeneous.

Because the main focus of the paper is on how to integrate DS theory and its developments into an XML framework and how to merge XML documents that involve uncertainties in the format of mass functions, we did not include research results that justify the propagations and combinations of mass functions reported in the paper. These results can be found in research papers such as [LGS86, SSM87, LHA03] etc. Instead, we emphasized on how such information, when encoded into XML structures, can be merged and how this procedure can be formally described in logical terminologies and then be executed as Prolog predicates. Developing this framework has involved a number of design decisions that we believe have resulted in a set of definitions for handling and merging uncertain information in a viable and useful way. Whilst other authors have considered some of these goals, we have shown in Section 5 how our approach is superior.

References

[CL96] J Cowie and W Lehnert. Information extraction. *Communications of the ACM*, 39:81–91, 1996.

[DP88] D Dubois and H Prade. *Possibility theory: An approach to the computerized processing of uncertainty*. Plenum Press, 1988.

[DP98] D Dubois and H Prade, editors. *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Volume 3. Kluwer, 1998.

[HS04] A Hunter and R Summerton. Fusion rules for context-dependent aggregation of structured news reports. *Journal of Applied Non-Classical Logics*, 14(3):329-366, 2004.

[HL04a] A Hunter and W Liu. Fusion rules for merging uncertain information. *Information Fusion Journal*, to appear, 2004.

```
⟨probability⟩
 ⟨prob value = “0.8”⟩
  ⟨possible values⟩
    ⟨name⟩
      ⟨probability⟩
        ⟨prob value = “1.0”⟩
          ⟨possible values⟩
            ⟨nameValue⟩John⟨/nameValue⟩
          ⟨/possible⟩
        ⟨/prob⟩
      ⟨/probability⟩
    ⟨/name⟩
  ⟨/possible⟩
  ⟨possible values⟩
    ⟨tel⟩
      ⟨probability⟩
        ⟨prob value = “1.0”⟩
          ⟨possible values⟩
            ⟨telNumber⟩1111⟨/telNumber⟩
          ⟨/possible⟩
        ⟨/prob⟩
      ⟨/probability⟩
    ⟨/tel⟩
  ⟨/possible⟩
⟨/prob⟩
⟨prob value = “0.2”⟩
  ⟨possible values⟩
    ⟨name⟩
      ⟨probability⟩
        ⟨prob value = “1.0”⟩
          ⟨possible values⟩
            ⟨nameValue⟩John⟨/nameValue⟩
          ⟨/possible⟩
        ⟨/prob⟩
      ⟨/probability⟩
    ⟨/name⟩
  ⟨/possible⟩
  ⟨possible values⟩
    ⟨tel⟩
      ⟨probability⟩
        ⟨prob value = “1.0”⟩
          ⟨possible values⟩
            ⟨telNumber⟩2222⟨/telNumber⟩
          ⟨/possible⟩
        ⟨/prob⟩
      ⟨/probability⟩
    ⟨/tel⟩
  ⟨/possible⟩
 ⟨/prob⟩
⟨/probability⟩
```

```
⟨probability⟩
 ⟨prob value = “0.8”⟩
  ⟨multiitem⟩
    ⟨name⟩John⟨/name⟩
    ⟨tel⟩1111⟨/tel⟩
  ⟨/multiitem⟩
 ⟨/prob⟩
 ⟨prob value = “0.2”⟩
  ⟨multiitem⟩
  ⟨name⟩John⟨/name⟩
    ⟨tel⟩2222⟨/tel⟩
  ⟨/multiitem⟩
 ⟨/prob⟩
⟨/probability⟩
```

Figure 2: Two probabilistic XML documents with the same information

[HL04b] A Hunter and W Liu. Structured Scientific Knowledge in XML: modelling, reasoning, merging, and querying. Technical Report, Department of CS, UCL. 2004.

[Hun02] A Hunter. Logical fusion rules for merging structured news reports. *Data and Knowledge Engineering*, 42:23–56, 2002.

[Hun02b] A Hunter. Merging structured text using temporal knowledge. *Data and Knowledge Engineering*, 41:29–66, 2002.

[Hun03] A Hunter. Evaluating the significance of inconsistency. *Proceedings of the International Joint Conference on AI (IJCAI'03)*, 468–473, 2003.

[KKA05] M van Keulen, A de Keijzer and W Alink. A probabilistic XML approach to data integration. *Proceedings of ICDE'05*, 2005.

[LGS86] J Lowrance, T Garvey and T Strat. A framework for evidential reasoning systems. *Proc. of National Conference on Artificial Intelligence (AAAI'86)*, 896-903, 1986.

[LHA03] W Liu, X Hong and K Adamson. Computational-workload based binarization and partition of qualitative Markov trees for belief combination. *Proceedings of the 7th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, LNAI 2711:306-318, Springer, 2003.

[LH+93] W Liu, J Hong, M McTear and J Hughes. An extended framework for evidential reasoning systems. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 7(3), 441-457, 1993.

[NJ02] A Nierman and H Jagadish. ProTDB: Probabilistic data in XML. In *Proceedings of the International Conference on Very Large Databases (VLDB'02)*, LNCS 1590: 646–657, Springer, 2002.

[Sha76] G Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[Sme88] Ph Smets. Belief functions. In Ph Smets, A Mamdani, D Dubois, and H Prade, editors, *Non-Standard Logics for Automated Reasoning*: 253–286. Academic Press, 1988.

[SSM87] G Shafer, P Shenoy, and K Mellouli. Propagating belief functions in qualitative Markov trees. *Int. J. of Approx. Reasoning*, 1:349-400, 1987.