Decision Support and Knowledge Management in Oncology using Hierarchical Classification

Mathieu D'Aquin, Sébastien Brachais, Jean Lieber, Amedeo Napoli

Decision Support and Knowledge Management in Oncology using Hierarchical Classification

Mathieu d'Aquin, Sébastien Brachais, Jean Lieber and Amedeo Napoli
LORIA, UMR 7503 BP 239, CNRS, INRIA, Nancy Universities,
54 506 Vandœuvre-lès-Nancy
{daquin, brachais, lieber, napoli}@loria.fr

Abstract. This paper presents the KASIMIR research project for the management of decision protocols in oncology. A decision protocol is a kind of decision tree implemented in an object-based representation formalism. A reasoner based on such a formalism and on hierarchical classification is coupled with a knowledge editor. This association provides an assistance for editing and maintenance of protocols, enabling the detection of errors and the comparison between versions of the protocol. In this way, a management of protocols takes fully advantage of the underlying knowledge representation and reasoning tools. This straightforward use of the protocol may be insufficient in some situations. Then, the protocol may have to be adapted for these situations. A study of protocol adaptation is presented. In particular a reasoner based on a combination of hierarchical classification and fuzzy logic is introduced.

1 Introduction

In this paper, we present a multidisciplinary research project on decision support and knowledge management in oncology. This research project relies on the development of the KASIMIR system in which experts in oncology, physicians, psycho-ergonomists and computer scientists participate. The first oncology domain that has been deeply studied within this research project considers breast cancer, whose treatment is based on a decision protocol (a kind of decision tree for decision support). In this paper, we are mainly interested in the computer science aspects of the KASIMIR research project, and especially on the knowledge representation and reasoning requirements for the management of protocols, among which editing, diffusion and maintenance of the protocols. The principles, methodologies and techniques that have been chosen for this research are those of object-based knowledge representation formalisms [14] (that are similar to description logics [1]) and case-based reasoning [12, 9].

In section 2, an overview of the KASIMIR research project is presented, which emphasises the use of two reasoning processes: straightforward application of the protocol and adaptation of the protocol. Section 3 presents the reasoning module of the KASIMIR system and its knowledge representation formalism [14]. Section 4 shows the benefit of this reasoning for assisting the editing and the maintenance of knowledge. Section 5 presents ongoing research on protocol adaptations in situations for which the straightforward application of the protocol raises difficulties. It presents a working extension of the KASIMIR reasoner based on fuzzy logic and outlines some other research about protocol adaptation. The paper ends with a conclusion including a discussion on future work.

2 The KASIMIR research project: Knowledge Management in Oncology

In this section we propose an overview of the KASIMIR system, whose objective is decision support and knowledge management for the treatment of cancer (see [10], for a more detailed overview). This system is currently under development within a multidisciplinary research project in which participate researchers in computer science (the authors of this paper), and in ergonomics (from the *Laboratoire d'ergonomie du* CNAM, Paris), experts in oncology (from the *Centre Alexis Vautrin*, Vandœuvre-lès-Nancy) and ONCOLOR, an association gathering together physicians from Lorraine involved in oncology.

Some protocols, similar to medical guidelines, are available for solving decision problems in oncology. These protocols are built according to evidence-based medicine principles [5, 22]. In the KASIMIR system, several of these protocols have been implemented (e.g., post-therapeutic surveillance for breast and prostate cancers) and the acquisition and implementation of other protocols are currently under development. One of these protocols is devoted to the treatment of breast cancer without metastasis and, in the rest of the paper, we simply mention it as "the" protocol. The KASIMIR system proposes treatments on the basis of the protocol. Its implementation relies on an object-based representation formalism and on hierarchical classification (see, e.g. [14] and [1] respectively).

For most cases (about 70%), a straightforward application of the protocol is sufficient: it provides a solution (e.g., a treatment) that can be directly reused. A case from the other 30% of cases is an "out of protocol" case. There are two kinds of out of protocol cases studied in the project. First, the cases for which the protocol does not provide a treatment (e.g., when the patient is a man: the protocol has not been written for men, though some men suffer from breast cancer). Second, the cases for which the proposed solution raises some difficulties (contraindication, impossibility of applying completely a treatment, etc.). For any kind of out of protocol cases, oncologists try to *adapt* the protocol during meetings of the so-called breast therapeutic decision committee (BTDC) that gathers together experts of the domains linked with breast oncology (e.g., chemotherapy, radiotherapy and surgery).

The adaptation is currently studied from the ergonomics and computer science viewpoints. For the latter, the design and the development of a reasoner based on adaptation is under study and implementation [11]. The design of this reasoner is based on case-based reasoning principles (CBR [18]): it has to select a patient class and adapt the treatment associated with this class in order to suggest a treatment for the current patient.

The adaptations can be used to propose evolutions of the protocol thanks to its confrontation with real cases. The idea is then to make suggestions of protocol evolutions based on frequently performed adaptations. This has been studied by psycho-ergonomists [21]. The implementation of a computer system that could make such suggestions is a long term future work.

The KASIMIR system is planned to be used by physicians of ONCOLOR in their daily practice. A validation study of KASIMIR has already been carried out [19]. The objective of this study was to see whether the use of the system improves health care quality. A set of 30 physicians had to propose treatments for patients with the help of the protocol in its paper form (the protocol is drawn as a decision tree associated to explanations in plain text, for being more easily readable and understandable) and/or with the help of KASIMIR. A statistically significant improvement of the compliance with the medical standards thanks to KASIMIR has been shown [19]. The use of KASIMIR should therefore improve the health care quality according to the paradigm of evidence-based medicine.

3 Reasoning and Knowledge Representation in KASIMIR

This section presents the part of the KASIMIR system based on a straightforward application of the protocol. It introduces the architecture and a description of the knowledge representation formalism and of the reasoner.

3.1 *Architecture of* KASIMIR

The development of the KASIMIR system is based on *genericity*, so that the customisation of this system for different protocols is as simple as possible. The principle of this genericity is that the knowledge base and the specification of the user interface are described in a set of XML files. These files are loaded into the KASIMIR reasoner. Any change or update only requires to add XML files with the associated modifications. Figure 1 presents the global architecture, while the user interface for querying the reasoner and displaying the results is given at figure 2.

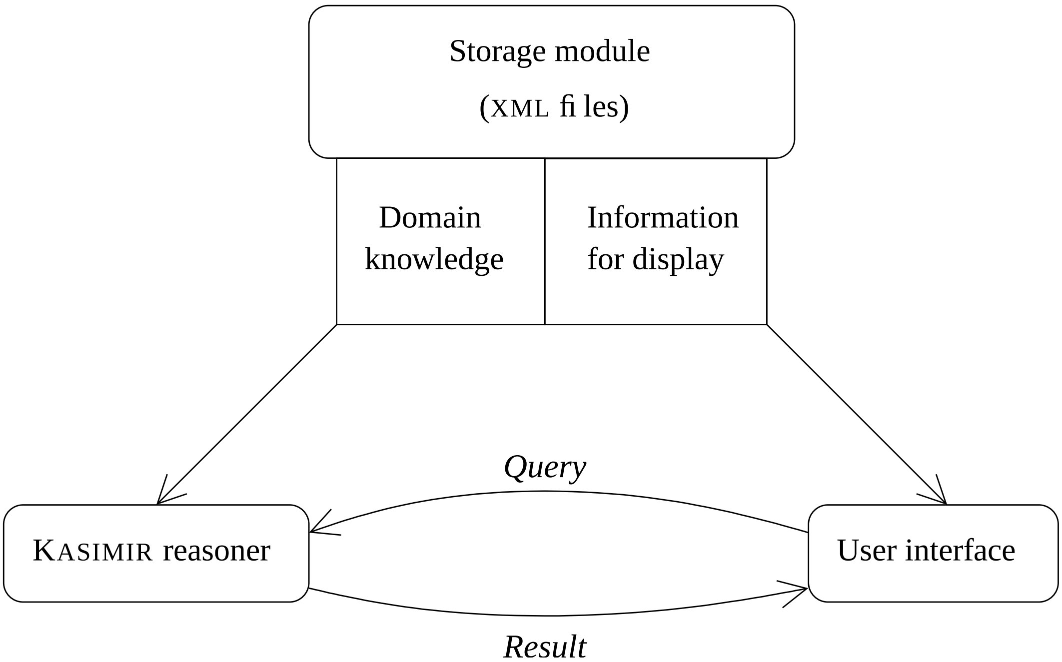

Figure 1: The current architecture of the KASIMIR system.

3.2 *The Knowledge Representation Formalism and the Reasoner*

Knowledge representation is at the heart of the knowledge management process in the KASIMIR system. It relies on an object-based representation system [14], that can be likened to a description logic system [1]. The basic representation unit is the *concept*, that represents a set of objects, or individuals, sharing a number of *properties*, or *attributes*. The set of objects is called the *extension* of the concept, while the corresponding set of properties is called the *intension* of the concept. An individual being a member of the extension of a concept is also called an *instance* of the concept. An attribute has a *domain*, i.e. the concept to which it is attached, and a *range*, determining the type of the admissible values of the attribute. The range of an attribute may be a primitive type (number, string, etc.) or another concept. In the latter case, the attribute defines a *relation* between its domain concept and its range concept.

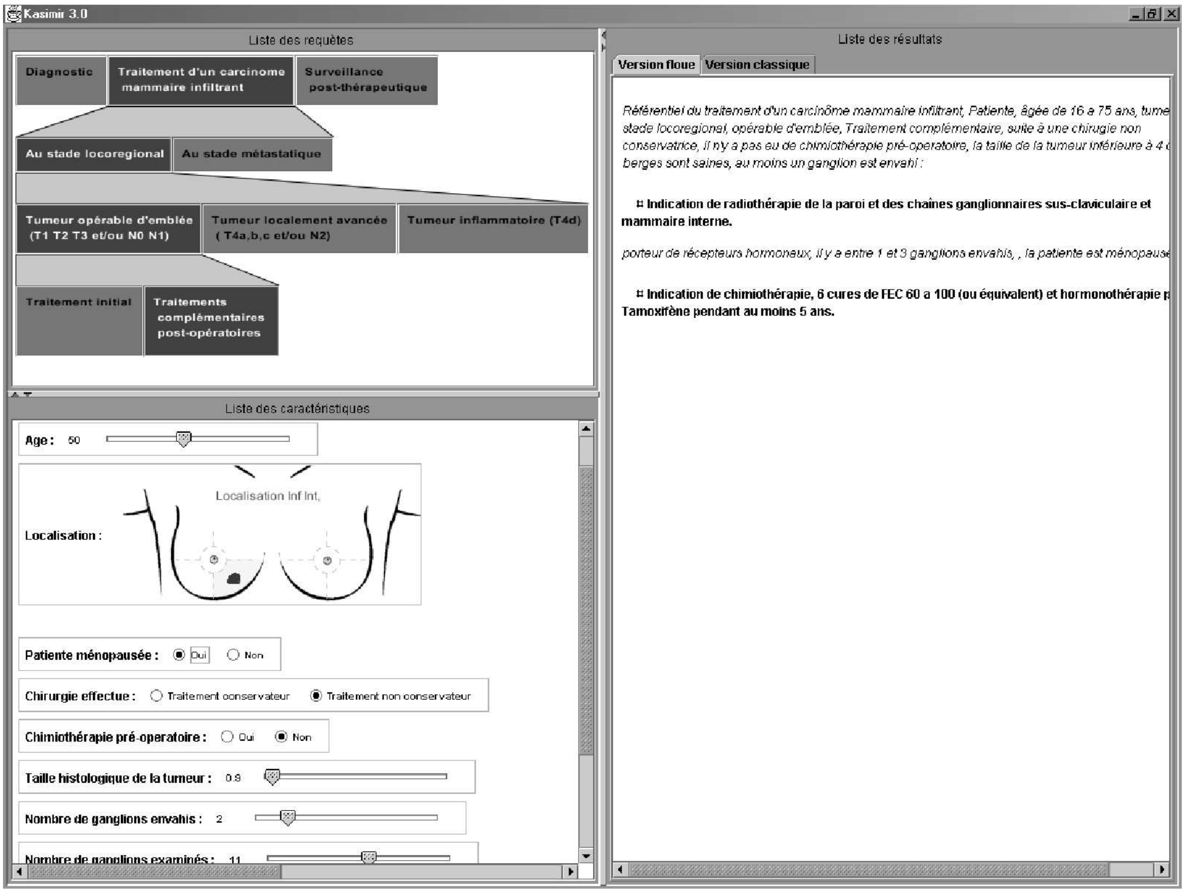

Figure 2: The KASIMIR user interface. The top left panel is used to choose a sub-protocol of the selected protocol (e.g., initial treatment, post-surgery treatment, etc.). The bottom left panel is used to enter characteristics of the patient (age, etc.) and of the tumour (size, etc.). The right panel is used to display treatment propositions together with some explanations; it is updated when any modification of the left panels is performed by the user.

Two kinds of concepts can be distinguished. *Primitive concepts* are considered as atoms of the representation system. They are used as building blocks for the *defined concepts*. Moreover, the intension of a primitive concept is empty, i.e. it has no attribute, while the intension of a defined concept is composed of attributes acting as a set of necessary and sufficient conditions for recognising an individual as an instance of the corresponding defined concept. The quality of the attributes of a defined concept, i.e. being necessary and sufficient conditions, is the basis for the concept classification process that is made precise below.

A *subsumption* relation ($\sqsubseteq$) is defined on the set of concepts in the following way: a concept $\mathtt{C}_1$ is *subsumed* by a concept $\mathtt{C}_2$, denoted by $\mathtt{C}_1 \sqsubseteq \mathtt{C}_2$, whenever the extension of $\mathtt{C}_1$ is necessarily included in the extension of $\mathtt{C}_2$, i.e. the concept $\mathtt{C}_1$ is more specific than the concept $\mathtt{C}_2$, or in a dual way, $\mathtt{C}_2$ is more general than $\mathtt{C}_1$. The subsumption relation is a partial ordering (based on inclusion of extensions) that organises concepts within a *hierarchy*, i.e. an acyclic directed graph denoted by $\mathcal{H}_{\mathcal{C}}$, where the subsumption relation is declared for primitive concepts, while it is calculated as follows for defined concepts. Given two defined concepts $\mathtt{C}_1$ and $\mathtt{C}_2$, the relation $\mathtt{C}_1 \sqsubseteq \mathtt{C}_2$ holds if and only if, for all attribute $\mathtt{a}_i$ in the subsuming concept $\mathtt{C}_2$, there exists a corresponding attribute $\mathtt{a}_j$ in the concept $\mathtt{C}_1$ that has the same name and whose characteristics verifies the constraints associated with $\mathtt{a}_i$ in $\mathtt{C}_2$. These constraints are relative to the range of the attribute and are verified in the following way:

- If the range of $\mathtt{a}_i$ in $\mathtt{C}_2$ is a primitive type, say $\mathtt{T}_2$, then the range of $\mathtt{a}_j$ in $\mathtt{C}_1$ must be a primitive type, say $\mathtt{T}_1$, equal to $\mathtt{T}_2$ or a subtype of $\mathtt{T}_2$.
- If the range of $\mathtt{a}_i$ in $\mathtt{C}_2$ is a concept, say $\mathtt{D}_2$, then the range of $\mathtt{a}_j$ in $\mathtt{C}_1$, say $\mathtt{D}_1$, must be subsumed by $\mathtt{D}_2$: $\mathtt{D}_1 \sqsubseteq \mathtt{D}_2$.
- If the range of $\mathtt{a}_i$ in $\mathtt{C}_2$ is an interval of numbers, say $[\mathtt{p}_2, \mathtt{q}_2]$, then the range of $\mathtt{a}_j$ in $\mathtt{C}_1$ must be an interval of numbers, say $[\mathtt{p}_1, \mathtt{q}_1]$, included in $[\mathtt{p}_2, \mathtt{q}_2]$.

Given the subsumption relation between concepts, the *classification process* applies to *concept classification* and *instance classification*:

- Concept classification is used for comparing defined concepts, and placing a new concept C in the concept hierarchy, under its most specific subsumers and over its most general subsumees [15, 1].
- Instance classification is used for recognising that an individual is an instance of a concept.

The example below illustrates the notions presented before. First, let us consider the following introductions of primitive concepts ($\dot\sqsubset$ is used for primitive concepts introduction):

$$\mathtt{any\text{-}localisation} \;\dot\sqsubset\; \top \qquad\qquad \mathtt{internal} \;\dot\sqsubset\; \mathtt{any\text{-}localisation}$$

$\top$ is the top concept, i.e. its extension contains all the individuals; `any-localisation` and `internal` are two primitive concepts and the latter is declared to be subsumed by the former. In a similar way, the following primitive concepts are introduced:

$$\mathtt{any\text{-}sex} \;\dot\sqsubset\; \top \qquad\qquad \mathtt{female} \;\dot\sqsubset\; \mathtt{any\text{-}sex} \qquad\qquad \mathtt{male} \;\dot\sqsubset\; \mathtt{any\text{-}sex}$$

`any-sex`, here, stands for “female or male”. It has been introduced in particular to specify the maximal range of the attribute `sex` in defined concepts (see below).

The defined concepts hereafter denote respectively the set of internal tumours of size S such that $0 \leq S < 4$ (in centimeters) and the set of women with an age between 40 and 80 having such a tumour ($\doteq$ is used for introducing defined concepts thanks to a conjunction of necessary and sufficient conditions and $\sqcap$ stands for the conjunction operator):

$$\mathtt{small\text{-}int\text{-}tumour} \doteq (\mathtt{size} : [0;4[\ \sqcap\ \mathtt{localisation} : \mathtt{internal})$$

$$\mathtt{WA}_{40-80}\mathtt{SIT} \doteq (\mathtt{sex} : \mathtt{female} \sqcap \mathtt{age} : [40,80] \sqcap \mathtt{tumour} : \mathtt{small\text{-}int\text{-}tumour})$$

`size`, `localisation`, `sex`, `age` and `tumour` are five attributes; their respective ranges in these concepts are the interval of real numbers $[0;4[$, the primitive concept `internal`, the primitive concept `female`, the interval of integers $[40,80]$ and the defined concept `small-int-tumour`.

Defined concepts are used for representing classes of patients sharing common characteristics: these classes are considered as “problems” to which “solutions” may be attached, provided that the class is in accordance with a given protocol. Actually, a solution corresponds to a specific cancer treatment, that can be applied to every individual in the class. Following this idea, the problem of finding a “solution” for a given “problem”, i.e. finding the right treatment for a given patient, is considered as a decision-support task, and relies on the classification process, as explained hereafter.

In the KASIMIR system, a protocol can be seen as a set of rules $\mathtt{R} = (\mathtt{pb} \longrightarrow \mathtt{Sol}(\mathtt{pb}))$, where $\mathtt{pb}$ denotes a problem and $\mathtt{Sol}(\mathtt{pb})$ a solution of $\mathtt{pb}$. A problem is a kind of concept, denoting a set of patients. A solution denotes a treatment. The decision-support process relies on an inference rule, that can be likened to the *modus ponens*, and that can be read as follows: whenever a problem $\mathtt{pb}_2$ is more general than a problem $\mathtt{pb}_1$, then every solution of $\mathtt{pb}_2$ is also a solution of $\mathtt{pb}_1$ [12]. Replacing $\mathtt{pb}_1$ with the problem to be solved denoted by $\mathtt{tgt}$, and $\mathtt{pb}_2$ with $\mathtt{pb}$, we obtain the inference rule:

$$\frac{\mathtt{tgt} \qquad \mathtt{pb} \sqsupseteq \mathtt{tgt} \qquad \mathtt{Sol}(\mathtt{pb}) \text{ is a solution of } \mathtt{pb}}{\mathtt{Sol}(\mathtt{pb}) \text{ is a solution of } \mathtt{tgt}} \tag{1}$$

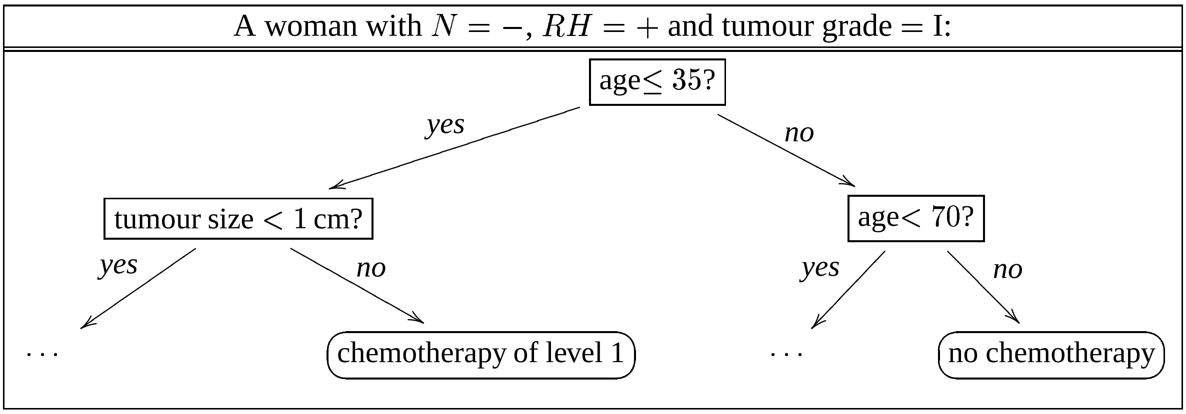

Figure 3: A decision tree (extracted and adapted from the breast cancer protocol described in `www.oncolor.org/`).

For example, let us consider the decision tree of figure 3. The concepts involved in this decision tree can be represented in the following way:

$\mathtt{any\text{-}boolean} \mathrel{\dot\sqsubset} \top$	$\mathtt{any\text{-}grade} \mathrel{\dot\sqsubset} \top$	$\mathtt{any\text{-}treatment} \mathrel{\dot\sqsubset} \top$
$\mathtt{true} \mathrel{\dot\sqsubset} \mathtt{any\text{-}boolean}$	$\mathtt{false} \mathrel{\dot\sqsubset} \mathtt{any\text{-}boolean}$	$\mathtt{grade\text{-}I} \mathrel{\dot\sqsubset} \mathtt{any\text{-}grade}$
$\mathtt{level\text{-}1\text{-}chemotherapy} \mathrel{\dot\sqsubset} \mathtt{any\text{-}treatment}$	$\mathtt{no\text{-}chemotherapy} \mathrel{\dot\sqsubset} \mathtt{any\text{-}treatment}$	

$$\mathtt{WN\_RH_+G_I} \doteq (\mathtt{sex} : \mathtt{female} \sqcap \mathtt{N} : \mathtt{false} \sqcap \mathtt{RH} : \mathtt{true} \sqcap \mathtt{tumour} : (\mathtt{grade} : \mathtt{grade\text{-}I}))$$

$$\mathtt{A}_{\leq 35} \doteq \mathtt{WN\_RH_+G_I} \sqcap (\mathtt{age} : [0, 35]) \qquad \mathtt{A}_{>35} \doteq \mathtt{WN\_RH_+G_I} \sqcap (\mathtt{age} : \,]35, +\infty[)$$

$$\mathtt{T}_{\geq 1} \doteq \mathtt{A}_{\leq 35} \sqcap (\mathtt{tumour} : (\mathtt{size} : [1; +\infty[\,)) \qquad \mathtt{A}_{\geq 70} \doteq \mathtt{A}_{>35} \sqcap (\mathtt{age} : [70, +\infty[\,)$$

The protocol rules $\mathtt{R} = (\mathtt{pb} \longrightarrow \mathtt{Sol}(\mathtt{pb}))$ are the following:

$$\mathtt{R}_1 = (\mathtt{T}_{\geq 1} \longrightarrow \mathtt{level\text{-}1\text{-}chemotherapy}) \qquad \mathtt{R}_2 = (\mathtt{A}_{\geq 70} \longrightarrow \mathtt{no\text{-}chemotherapy})$$

Applying the inference rule (1), the classification of the target problem $\mathtt{tgt}$, in the concept hierarchy considered as a problem hierarchy, returns the set of the problems subsuming $\mathtt{tgt}$. As soon as a subsuming problem has an associated solution, this solution can be reused in the context of $\mathtt{tgt}$. For example, consider the patient represented by the following target problem:

$$\begin{aligned}\mathtt{tgt} = (&\mathtt{sex} : \mathtt{female} \sqcap \mathtt{N} : \mathtt{false} \sqcap \mathtt{RH} : \mathtt{true} \\ &\sqcap \mathtt{age} : [75, 75] \sqcap \mathtt{tumour} : (\mathtt{grade} : \mathtt{grade\text{-}I}))\end{aligned}$$

The classification of `tgt` in the problem hierarchy shows that it is subsumed by the following concepts: $\mathtt{WN\_RH_{+}G_{I}}$, $\mathtt{A}_{>35}$ and $\mathtt{A}_{\geq 70}$. The solution `no-chemotherapy` is associated to the latter concept by the rule $\mathtt{R}_2$. Therefore, the KASIMIR system indicates that, for this patient, no chemotherapy is recommended.

The representation formalism presented above is rather simple (compared to a description logics such as RACER [7]), but it is sufficient for the protocols that has been represented so far in KASIMIR. One advantage of this simplicity is the low complexity in time of the inferences: in the worst case, the classification is in $O(n_a{\cdot}n_c)$ where n_a and n_c are the numbers of attributes and concepts (when the hierarchy is a well-balanced tree, this complexity is $O(n_a \cdot \log(n_c))$). In practice, even with the more complex protocol represented in the KASIMIR system (the protocol for the treatment of breast cancer without metastasis which includes about 1200 concepts and about 50 attributes), the result of the reasoning process is given on the spot on a current personal computer.

4 Editing and Maintenance of Knowledge

Representing a decision protocol in a knowledge representation formalism can become tedious, for a big knowledge base. Thus, raised the need for KASIMIR knowledge engineers of a knowledge editor. The PROTÉGÉ system [16] has been chosen for this purpose, in particular because many useful and available tools have been integrated into its architecture. PROTÉGÉ has been customised to become a knowledge editor for KASIMIR and has been connected to the KASIMIR reasoner. We present in the following the PROTÉGÉ system in the framework of the KASIMIR system.

4.1 Using PROTÉGÉ *and* KASIMIR *for Knowledge Editing and Visualisation*

In this section, an overview of the connection of PROTÉGÉ and KASIMIR is presented. This connection is detailed in [3].

The first step of the customisation of PROTÉGÉ for KASIMIR knowledge base editing has been to integrate the KASIMIR knowledge representation model into PROTÉGÉ. Then, a knowledge base relative to a protocol can be edited and then exported as a KASIMIR knowledge base. Furthermore, the knowledge editor has been connected to the reasoner of the KASIMIR system. This enables in particular to detect errors during knowledge editing sessions. For example, it may occur that two problems $\mathtt{pb}_1$ and $\mathtt{pb}_2$ are edited with two equivalent definitions (i.e., they denote the same set of individuals: $\mathtt{pb}_1 \sqsupseteq \mathtt{pb}_2$ and $\mathtt{pb}_2 \sqsupseteq \mathtt{pb}_1$); in such a situation, the reasoner detects this equivalence and the user is alerted that only one problem definition is useful. Another example is when the *declared* hierarchy in PROTÉGÉ does not match in a one to one correspondence with the *calculated* hierarchy of KASIMIR: this mismatch usually means that there is an editing error. In practice, these warnings have proven to be useful to detect, at an early stage, many editing errors.

Two visualisation modules have been integrated in PROTÉGÉ allowing the display of the KASIMIR hierarchy of problems from the protocol being edited: PALÉTUVIER and HYPERTREE (see figure 4). The combined use of these two visualisation modules and of the classical tree widget of PROTÉGÉ provides several useful features for hierarchy visualisation such as navigation, global or focused view.

Another module for knowledge maintenance is described with more details below.

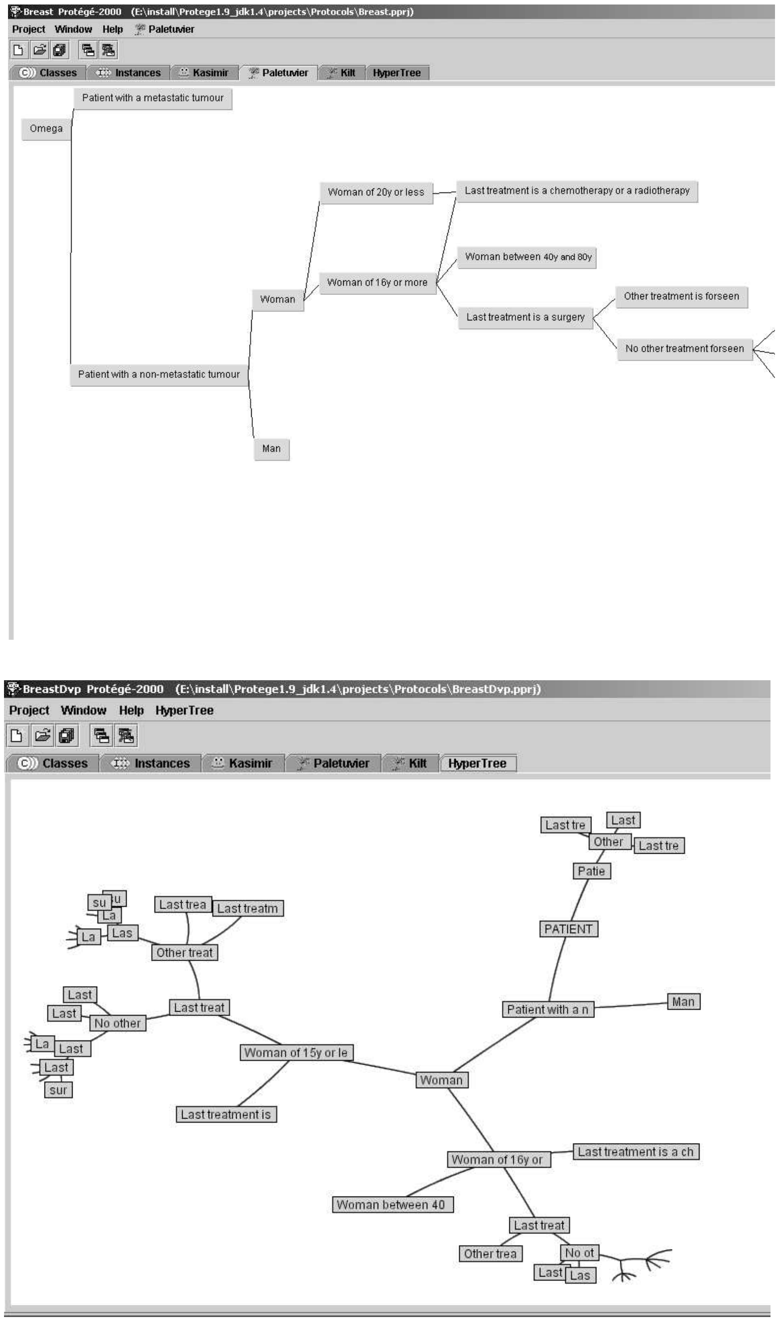

Figure 4: Two visualisation modules: PALÉTUVIER (at the top) and HYPERTREE (at the bottom). PALÉTUVIER can help, in particular, for visualising multiple inheritance hierarchies, without node duplications. HYPERTREE is a free API [2] for hyperbolic tree visualisation and provides a good support for navigation through hierarchies, thanks to its "fish-eye" effect.

4.2 KILT*: a Module for Comparing Knowledge Base Versions*

During an update of a KASIMIR knowledge base, the need for automatically comparing the old base $\mathtt{KB}_{\mathtt{before}}$ (before the update) and the new base $\mathtt{KB}_{\mathtt{after}}$ (after the update) has appeared. A module comparing versions has to highlight what has been actually updated, to check whether the modifications are in accordance with the intents of the knowledge engineer. The comparison module, called KILT, has been implemented and integrated into PROTÉGÉ. KILT builds a partition of the problems represented in $\mathtt{KB}_{\mathtt{before}}$ and/or $\mathtt{KB}_{\mathtt{after}}$ in four parts (recall that a problem is described by a concept denoting a set of patients, and is possibly associated with a solution, i.e. a treatment):

1. The problems that appear in the two bases, with the same solutions;
2. The problems that appear in the two bases, with different solutions;
3. The obsolete problems, appearing in $\mathtt{KB}_{\mathtt{before}}$ but not in $\mathtt{KB}_{\mathtt{after}}$;
4. The new problems, appearing in $\mathtt{KB}_{\mathtt{after}}$ but not in $\mathtt{KB}_{\mathtt{before}}$.

From an algorithmic point of view, it is easy to make a partition of the different problems in this way, thanks to the use of the KASIMIR reasoner. For example, the new problems in category (4) can be found in the following way. Each problem $\mathtt{pb}_{\mathtt{after}}$ of $\mathtt{KB}_{\mathtt{after}}$ is classified in the hierarchy of $\mathtt{KB}_{\mathtt{before}}$, which enables to check whether there is a problem $\mathtt{pb}_{\mathtt{before}}$ of $\mathtt{KB}_{\mathtt{before}}$ that is equivalent to $\mathtt{pb}_{\mathtt{after}}$, i.e. $\mathtt{pb}_{\mathtt{after}} \sqsubseteq \mathtt{pb}_{\mathtt{before}}$ and $\mathtt{pb}_{\mathtt{before}} \sqsubseteq \mathtt{pb}_{\mathtt{after}}$. If this is not the case, then $\mathtt{pb}_{\mathtt{after}}$ is a new problem. The three other categories of problems –(1), (2) and (3)– can be found in a similar way. This shows that the implementation of KILT is easy, once the connection with a reasoner like the KASIMIR reasoner, is done.

This partition can be visualised using the hierarchy visualisation module PALÉTUVIER (see section 4.1), with a different colour for each type of problem (see figure 5).

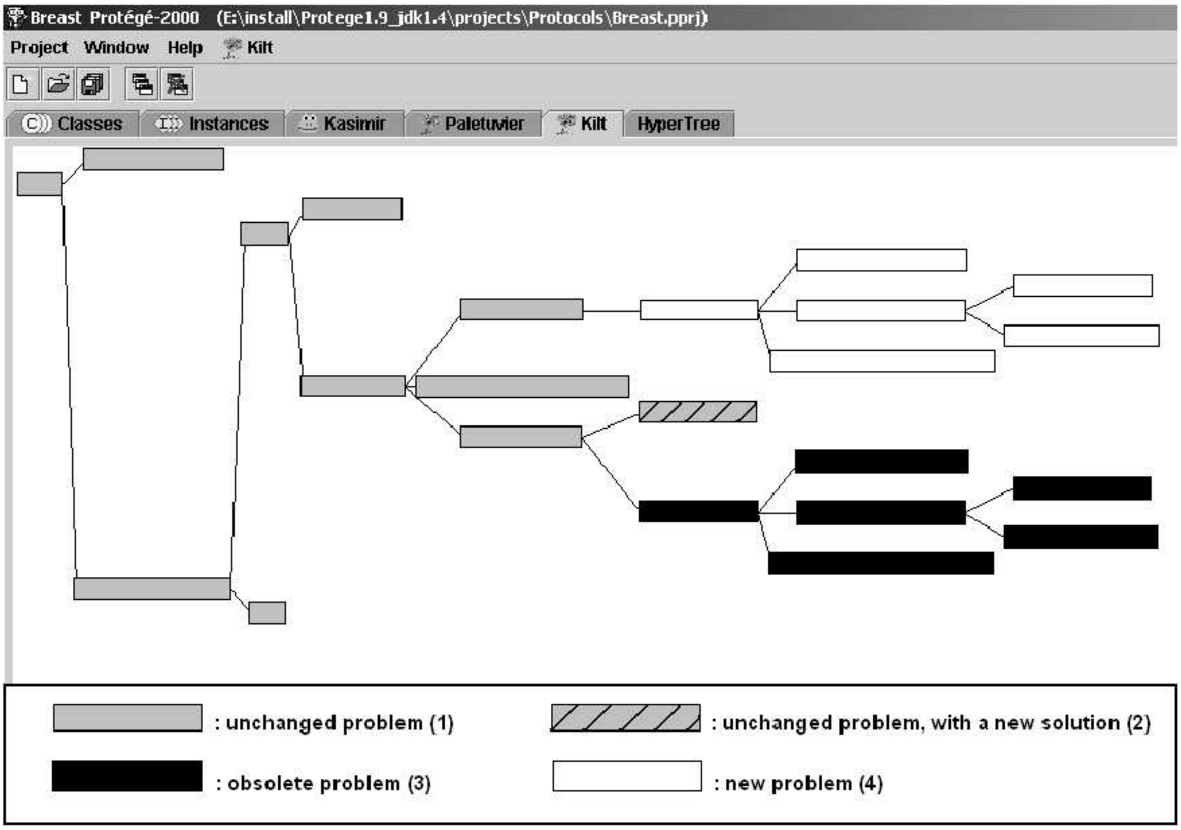

Figure 5: Visualisation of a coloured hierarchy of problems (each problem is coloured according to its status wrt the old knowledge base and the new one).

KILT is used in PROTÉGÉ in the following way. During a session, $\mathtt{KB}_{\mathtt{before}}$ corresponds to the state of the knowledge base at the beginning of the session, and $\mathtt{KB}_{\mathtt{after}}$ to its current state. Therefore, the KILT module enables to visualise the editing modifications, i.e. addition or removal of a problem, and association of another solution to an already known problem, at any time of the session.

KILT can be compared to PROMPTDIFF, an algorithm for comparing ontology versions, based on a set of matching algorithms (called *matchers*) [17]. Both tools enable to differentiate what has changed from what has not changed in two versions of a knowledge base. The main difference between PROMPTDIFF and KILT is that the former is based on a purely syntactic approach, whereas the latter is based on the semantics of the knowledge units that are manipulated. More precisely, all the PROMPTDIFF matchers described in [17] work at a syntactic level: they are based either on the tree structure of the two ontology versions or on the names of the slots and classes. By contrast, KILT performs comparisons at a semantic level: two concepts match when they have equivalent definitions, based on their attribute values and on the subsumption relation between classes. The main drawback of KILT is that it assumes that the attributes –and their names– do not change from one knowledge base version to another, whereas PROMPTDIFF can match slots having different names. On the other hand, if two concepts are matched by KILT, whatever their names or their positions in their respective hierarchies are, they are proven to be equivalent, whereas the PROMPTDIFF matchers are based on heuristics. An interesting study would be to combine KILT and PROMPTDIFF, using, for example, KILT as one of the PROMPTDIFF matchers.

5 Adaptation in KASIMIR

As said above, in section 2, during the meetings of the breast therapeutic decision committee (BTDC) the protocol is *adapted* for the “out of protocol” cases. In order to assist experts of the BTDC, the modelling of this kind of inference is currently studied, for building a protocol adaptation reasoner. For this purpose, case-based reasoning principles are used and, in particular, the notions of similarity paths and reformulations [13]. In this section, a study on adaptation knowledge acquisition from experts is briefly presented and then, the description of a first version of an adaptation reasoner, taking into account the threshold effect is described.

5.1 Adaptation Knowledge Acquisition from Experts

Minutes of BTDC meetings have been recorded and analysed by a psycho-ergonomist [21]. Thanks to these minutes, adaptation knowledge acquisition has been carried out, with the psycho-ergonomist, the experts in oncology and the computer science specialists. In [11], this adaptation knowledge acquisition is described. In particular, several general schemas of adaptation knowledge are presented together with the needs involved in knowledge representation. Some of these acquired schemas are summarised below:

- A first adaptation schema can be applied when data about the patient are missing. In such a situation, the so-called Wald pessimistic criterion [24, 4] can be applied: the decision (treatment) is chosen on the basis of its worst consequences (that must be avoided).

- A second adaptation schema can be applied when an element of the treatment proposed by the protocol is contraindicated. Then, this element must be substituted by another treatment element having similar expected benefits but not the same undesirable effects.

- A third adaptation schema is detailed hereafter.

5.2 *Taking into account the Threshold Effect*

When a numerical patient characteristic (e.g., the age) is close to a decision threshold of the protocol (i.e., a bound of an interval), the straightforward application of the protocol raises a problem. For example, let $\mathtt{srce}_1$, $\mathtt{srce}_2$ and $\mathtt{tgt}$ be the following problems:

$$\begin{aligned} \mathtt{srce}_1 &= (\mathtt{sex : female} \sqcap \mathtt{tumour : (size : } [0; 4])) \\ \mathtt{srce}_2 &= (\mathtt{sex : female} \sqcap \mathtt{tumour : (size : }]4; 7])) \\ \mathtt{tgt} &= (\mathtt{sex : female} \sqcap \mathtt{age : } [56, 56] \sqcap \mathtt{tumour : (size : } [3.8; 3.8])) \end{aligned}$$

$\mathtt{srce}_1$ (resp., $\mathtt{srce}_2$) is assumed to be a problem of the protocol and $\mathtt{Sol}(\mathtt{srce}_1)$ (resp., $\mathtt{Sol}(\mathtt{srce}_2)$) is assumed to be the solution of $\mathtt{srce}_1$ (resp., of $\mathtt{srce}_2$) in the protocol. Moreover, it is assumed that $\mathtt{Sol}(\mathtt{srce}_1) \neq \mathtt{Sol}(\mathtt{srce}_2)$. $\mathtt{tgt}$ is a target problem. Answering the question "What solution should be associated with $\mathtt{tgt}$?" with a straightforward application of the protocol returns $\mathtt{Sol}(\mathtt{srce}_1)$ and not $\mathtt{Sol}(\mathtt{srce}_2)$, because $\mathtt{tgt} \sqsubseteq \mathtt{srce}_1$ and $\mathtt{tgt} \not\sqsubseteq \mathtt{srce}_2$. But, since the size of the tumour of the patient associated with $\mathtt{tgt}$, 3.8 cm, is close to the threshold 4 cm, this decision is not certain, for two reasons. First, the decision threshold of 4 cm is uncertain, second, the measure 3.8 cm may be imprecise. A better idea is to propose to the user of the KASIMIR system *both* solutions $\mathtt{Sol}(\mathtt{srce}_1)$ and $\mathtt{Sol}(\mathtt{srce}_2)$. A reasoner based on *fuzzy hierarchical classification* [9] and extending the reasoner described in section 3.2, has been developed for this purpose.

This reasoner is based on a combination of object-based representation system and fuzzy logic [20]. It relies on a *fuzzification* of the problems contained in the protocol. As presented in section 3.2, a problem $\mathtt{srce}$ of the protocol is described by a concept which denotes a *set* of individuals. The principle of the fuzzification is to transform $\mathtt{srce}$ in a *fuzzy problem* $\mathcal{F}\mathtt{srce}$, represented by a *fuzzy concept* which denotes a *fuzzy set* of individuals. This fuzzification of the protocol has been achieved, thanks to the help of an expert in oncology. For example, for the problems $\mathtt{srce}_1$ and $\mathtt{srce}_2$ above, the parts that have to be fuzzified are the thresholds 4 cm and 7 cm: a fuzzy zone of ± 1 cm is chosen. A linear by pieces fuzzy set membership function was used for fuzzifying the (classical) intervals $[0; 4]$ and $]4; 7]$, as shown on figure 6. These fuzzy intervals are denoted $\mathcal{F}[0; 4]$ and $\mathcal{F}[4; 7]$. Therefore, the fuzzified problems issued from $\mathtt{srce}_1$ and $\mathtt{srce}_2$ are:

$$\begin{aligned} \mathcal{F}\mathtt{srce}_1 &= (\mathtt{sex : female} \sqcap \mathtt{tumour : (size : } \mathcal{F}[0; 4])) \\ \mathcal{F}\mathtt{srce}_2 &= (\mathtt{sex : female} \sqcap \mathtt{tumour : (size : } \mathcal{F}[4; 7])) \end{aligned} \tag{2}$$

Technically, what is presented above must be made precise. The description that follows is inspired from [23]. A fuzzy concept denotes a fuzzy set of individuals. More precisely, an interpretation is a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the *interpretation domain* (a classical set) and $\cdot^{\mathcal{I}}$ is the *interpretation function*, mapping a fuzzy concept $\mathcal{F}\mathtt{C}$ in a fuzzy subset $\mathcal{F}\mathtt{C}^{\mathcal{I}} = \mu$ of $\Delta^{\mathcal{I}}$, i.e., a function $\mu : \Delta^{\mathcal{I}} \longrightarrow [0; 1]$. In the fuzzy concept definitions (e.g., the definitions of $\mathcal{F}\mathtt{srce}_1$ and $\mathcal{F}\mathtt{srce}_2$ in (2)), the conjunction operator $\sqcap$ must be read as a $\min$ (the Zadeh t-norm [20]). More precisely, let $\mathcal{F}\mathtt{C}_1$ and $\mathcal{F}\mathtt{C}_2$ be two concepts, $\mathcal{I}$ be an interpretation, $\mu_1 = \mathcal{F}\mathtt{C}_1^{\mathcal{I}}$ and $\mu_2 = \mathcal{F}\mathtt{C}_2^{\mathcal{I}}$. Then, $\mu = (\mathcal{F}\mathtt{C}_1 \sqcap \mathcal{F}\mathtt{C}_2)^{\mathcal{I}}$ can be defined by $\mu(x) = \min\{\mu_1(x), \mu_2(x)\}$ for each $x \in \Delta^{\mathcal{I}}$.

A difference from our approach and the approach of [23] lies in the subsumption between two fuzzy concepts. Indeed, in [23], the subsumption is a classical binary relation between fuzzy concepts: either it holds or it does not. By contrast, in our approach, the subsumption relation between concepts ($\sqsupseteq$) is fuzzified in a *fuzzy subsumption relation* $\mathcal{S}$, i.e. an asymmetric similarity measure that associates to two concepts $\mathtt{C}_1$ and $\mathtt{C}_2$ a degree $\mathcal{S}(\mathtt{C}_1, \mathtt{C}_2) \in [0; 1]$ indicating "how $\mathtt{C}_1$ subsumes $\mathtt{C}_2$".

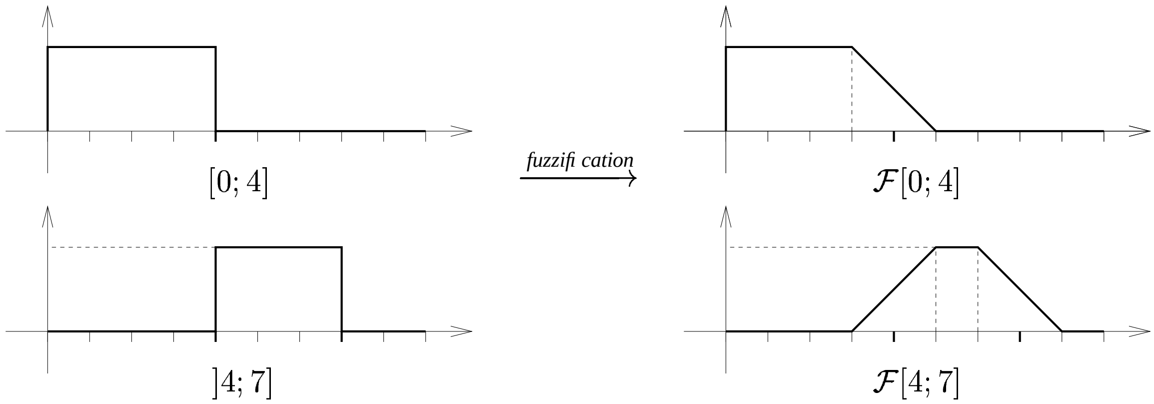

Figure 6: Fuzzification of Intervals.

The way $\mathcal{S}(\mathtt{C}_1, \mathtt{C}_2)$ is calculated depends on the nature of the concepts $\mathtt{C}_1$ and $\mathtt{C}_2$. We have not fuzzified the primitive concepts so far: if $\mathtt{C}_1$ and $\mathtt{C}_2$ are two primitive concepts, $\mathcal{S}(\mathtt{C}_1, \mathtt{C}_2)$ equals 1 if $\mathtt{C}_1 \sqsupseteq \mathtt{C}_2$ and 0 otherwise. For defined concepts $\mathtt{C}_1$ and $\mathtt{C}_2$, $\mathcal{S}(\mathtt{C}_1, \mathtt{C}_2)$ is the minimum of $\mathcal{S}_\mathtt{a}(\mathtt{C}_1, \mathtt{C}_2) = \mathcal{S}(\mathtt{C}_1.\mathtt{a}, \mathtt{C}_2.\mathtt{a})$ for the attributes $\mathtt{a}$ of $\mathtt{C}_1$, where $\mathtt{C.a}$ is the range of the attribute $\mathtt{a}$ in the concept $\mathtt{C}$. When the ranges of $\mathtt{a}$ in $\mathtt{C}_1$ and $\mathtt{C}_2$ are fuzzified numerical intervals μ_1 and μ_2, $\mathcal{S}_\mathtt{a}(\mathtt{C}_1, \mathtt{C}_2)$ is calculated by a fuzzification of the classical relation $\supseteq$ between intervals. In the KASIMIR fuzzy reasoner, we have chosen $\mathcal{S}_\mathtt{a}(\mathtt{C}_1, \mathtt{C}_2) = \min_{x \in \mathbb{R}} \min(1 + \mu_1(x) - \mu_2(x), 1)$, which is based on the fuzzification of the definition of $A \supseteq B$ by $\forall x, x \in B \Rightarrow x \in A$ and on the Lukasiewicz entailment $(u, v) \mapsto \min(1 - u + v, 1)$ [20]. From an implementation viewpoint, this can be easily calculated when μ_1 and μ_2 are linear by pieces.

Note that $\mathcal{S}$ is an extension of $\sqsupseteq$ in the sense that if $\mathtt{C}_1$ and $\mathtt{C}_2$ are two non fuzzy concepts, then $\mathtt{C}_1 \sqsupseteq \mathtt{C}_2$ iff $\mathcal{S}(\mathtt{C}_1, \mathtt{C}_2) = 1$. Let us resume the example above, with $\mathtt{srce}_1$ and $\mathtt{srce}_2$ fuzzified in $\mathcal{F}\mathtt{srce}_1$ and $\mathcal{F}\mathtt{srce}_2$, and $\mathtt{tgt}$ fuzzified in $\mathtt{tgt}$ ($\mathcal{F}\mathtt{tgt} = \mathtt{tgt}$: the value of $\mathtt{tgt}$ is assumed to be precise). It comes then that $\mathcal{S}(\mathcal{F}\mathtt{srce}_1, \mathtt{tgt}) = 0.6$ and $\mathcal{S}(\mathcal{F}\mathtt{srce}_2, \mathtt{tgt}) = 0.4$.

Furthermore, the relation "is a solution of" linking a problem and a solution is fuzzified: $\mathtt{Sol}(\mathtt{pb})$ solves $\mathtt{pb}$ with a truth value of $v \in [0, 1]$. v measures the confidence or the precision of $\mathtt{Sol}(\mathtt{pb})$ wrt $\mathtt{pb}$. $\mathtt{Sol}(\mathtt{pb})$ is said to be an s-solution of $\mathtt{pb}$ if $v \geq s$, with v, the truth value of "$\mathtt{Sol}(\mathtt{pb})$ solves $\mathtt{pb}$".

The reasoner manipulating the fuzzy concepts is based on the following inference rule that can be likened to the inference rule (1):

$$\frac{\mathtt{tgt} \quad \mathcal{S}(\mathtt{pb}, \mathtt{tgt}) = s \quad \mathtt{Sol}(\mathtt{pb}) \text{ is a solution of } \mathtt{pb}}{\mathtt{Sol}(\mathtt{tgt}) = \mathtt{Sol}(\mathtt{pb}) \text{ is an } s\text{-solution of } \mathtt{tgt}}$$

Therefore, in the example, both solutions $\mathtt{Sol}(\mathtt{srce}_1)$ and $\mathtt{Sol}(\mathtt{srce}_2)$ can be proposed to the user, with the respective confidence levels of 0.6 and 0.4.

In [9], the algorithm of fuzzy hierarchical classification is presented. It is based on a best-first search in the problem hierarchy $\mathcal{H}_\mathcal{C}$, with decreasing values of $\mathcal{S}(\mathtt{pb}, \mathtt{tgt})$, for $\mathtt{pb} \in \mathcal{H}_\mathcal{C}$.

The KASIMIR interface (see figure 2) has been adapted for displaying several propositions of solutions (see figure 7). The development of a simpler interface in planned. This interface would be closer to the one of figure 2 and would only present the solution with the higher score, but would point out the possible closeness to one or several decision threshold(s), as a warning.

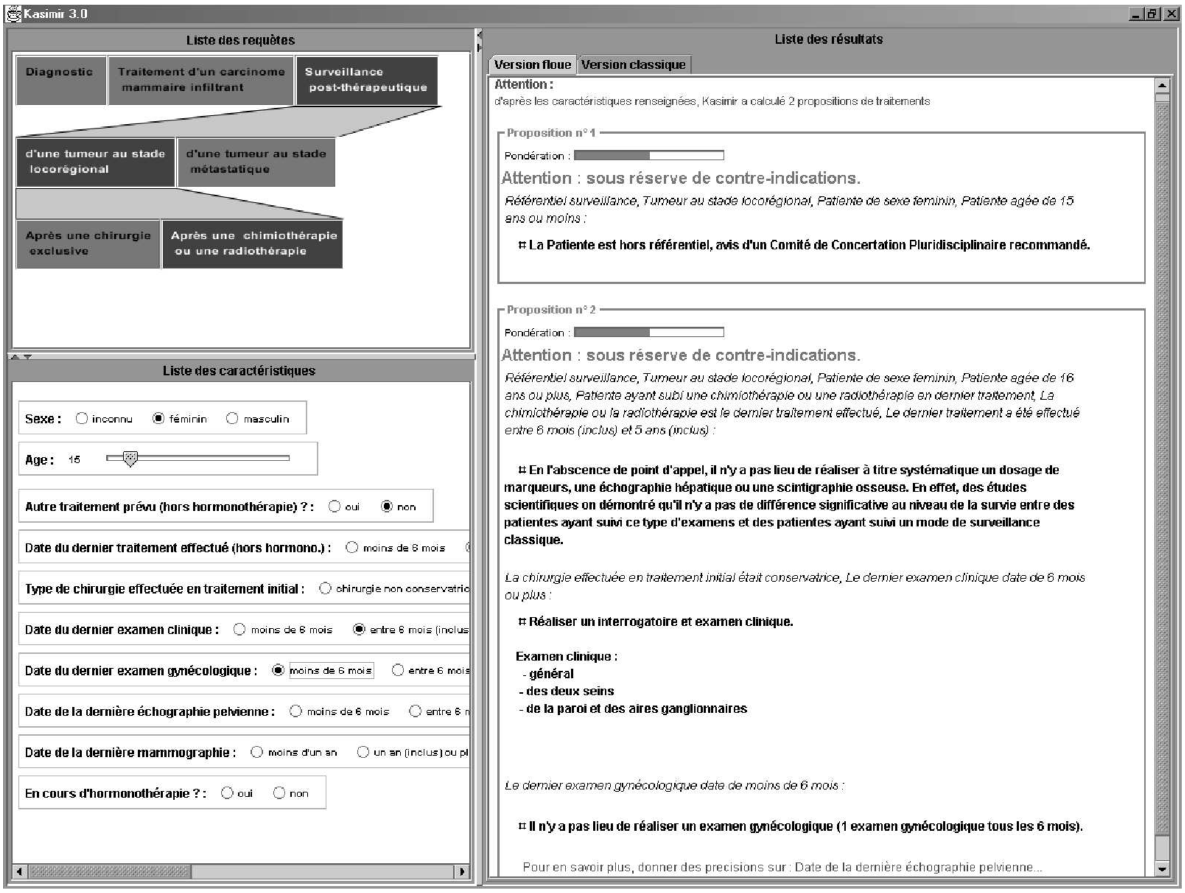

Figure 7: The KASIMIR interface for fuzzy application of the protocol.

Remark: The necessity to take into account the threshold effect has been involved by the adaptation knowledge acquisition described in [11]. But, up to case-based reasoning terminology, the approach that has been followed come under adaptation *and* retrieval steps of CBR. With the above example, the retrieval selects both cases $(\mathtt{srce}_1, \mathtt{Sol}(\mathtt{srce}_1))$ and $(\mathtt{srce}_2, \mathtt{Sol}(\mathtt{srce}_2))$, with a preference for the former (the fuzzifi ed subsumption relation is an asymmetric similarity measure). The adaptation of $(\mathtt{srce}_1, \mathtt{Sol}(\mathtt{srce}_1))$ to solve $\mathtt{tgt}$ is very simple in this situation: it is an adaptation by copy that gives $\mathtt{Sol}(\mathtt{srce}_1)$ as solution of $\mathtt{tgt}$, with the indicated confi dence level of 0.6. Current studies on adaptation for KASIMIR involves more complex adaptations, with solution modifi cations.

6 Conclusion and Future Work

The KASIMIR system is developed in the framework of the KASIMIR project whose goal is knowledge management in oncology. A part of this system is destined for the physicians and is constituted by an user interface, medical protocols represented in an object-based representation formalism and two reasoners: a reasoner based on hierarchical classifi cation and a reasoner based on fuzzy logic (other reasonings based on adaptation are currently studied). Another part of this system is destined for the knowledge engineers and contains several modules embedded in the PROTÉGÉ architecture and using the KASIMIR reasoner for the editing, visualisation and maintenance of knowledge. The example of KILT, a module for comparing two versions of a decision protocol has been detailed. One goal of this paper is to show that the technologies of knowledge representation and automatic reasoning are useful for maintenance of decision protocols.

The current research in computer science on the KASIMIR project follows two main directions. The objective of the first one is to embed the KASIMIR system in a semantic portal for oncology, i.e., a Web server relying on the principles and technologies of the semantic Web [6] in order to provide an intelligent access to knowledge and services that are useful for oncology. One of the main issues of the semantic Web relies on interoperability for knowledge and applications. Thus, building a semantic portal implies a standardisation for knowledge and software components of the KASIMIR system. For the knowledge bases, standardisation relies on a sharable domain model, and leads to the definition of general ontologies in oncology. This kind of knowledge base reengineering requires to replace the ad hoc knowledge representation formalism of KASIMIR with knowledge representation formalisms for the semantic Web such as OWL [25]. This evolution will allow the use of a wider set of knowledge representation primitives, such as disjunction of concepts for example. This work also implies a new software architecture, including the KASIMIR reasoner and the editing, visualisation and maintenance modules. This architecture must take into account constraints related to the distributed and dynamic environment of the semantic Web. A software architecture based on several Web services, implementing the KASIMIR modules and using standard Web services technologies [8] seems to be adapted.

The other research direction is about adaptation knowledge acquisition. Three approaches will be studied: automatic learning acquisition, acquisition from experts and a combination of both. This future work involves the study of the following questions in the field of knowledge representation for adaptation: “How can the acquired adaptation schemas be implemented?”, “What are the changes in problem and solution representations for adaptation compared to straightforward application of the protocol?” To answer the second question, a first study has shown that the composition of a treatment must be represented at different levels of details (e.g., for a chemotherapy, each of the drugs used must be represented, and not only the set of these drugs as a whole). Moreover, the expected benefits and the undesirable effects of a treatment have to be represented.

Acknowledgments

The authors would like to thank the other members of the KASIMIR project and, in particular, Julien Lévêque who has developed the reasoner that is based on fuzzy hierarchical classification and an interface allowing the visualisation of the results following such a classification process. They also would like to thank the referees for their comments.

References

[1] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, cambridge, UK, 2003.

[2] C. Bouthier. Hypertree Java Library. `http://hypertree.sf.net/`, 2003.

[3] M. d’Aquin, C. Bouthier, S. Brachais, J. Lieber, and A. Napoli. Knowledge Edition and Maintenance Tools for a Semantic Portal in Oncology. Research report, Loria, `http://www.loria.fr`, 2003.

[4] D. Dubois, H. Prade, and R. Sabbadin. Decision-theoretic foundations of qualitative possibility theory. *European Journal of Operational Research*, 128:459–478, 2001.

[5] Evidence-based medicine working-group. Evidence-based medicine. A new approach to teaching the practice of medicine. *JAMA*, 17:268, 1992.

[6] D. Fensel, J. Hendler, H. Lieberman, and W. Wahlster, editors. *Spinning the Semantic Web*. The MIT Press, Cambridge, Massachusetts, 2003.

[7] V. Haarslev and R. Möller. Description of the Racer System and its Applications. In C. Goble, D. L. McGuinness, R. Möller, and P. F. Patel-Schneider, editors, *Proceedings of the 2001 International Description Logics Workshop*, pages 46–55, 2001.

[8] http://www.w3.org/2002/ws. Web Services Activity. last consultation: September, 2003.

[9] J. Lieber. Strong, Fuzzy and Smooth Hierarchical Classification for Case-Based Problem Solving. In F. van Harmelen, editor, *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI-02), Lyon, France*, pages 81–85. IOS Press, Amsterdam, 2002.

[10] J. Lieber, M. d'Aquin, P. Bey, B. Bresson, O. Croissant, P. Falzon, A. Lesur, J. Lévêque, V. Mollo, A. Napoli, M. Rios, and C. Sauvagnac. The Kasimir Project: Knowledge Management in Cancerology. In *Proceedings of the 4th International Workshop on Enterprise Networking and Computing in Health Care Industry (HealthComm 2002)*, pages 125–127, 2002.

[11] J. Lieber, M. d'Aquin, P. Bey, A. Napoli, M. Rios, and C. Sauvagnac. Acquisition of adaptation knowledge for breast cancer treatment decision support. In Michel Dojat, Elpida Keravnou, and Pedro Barahona, editors, *9th Conference on Artificial Intelligence in Medicine in Europe2003 - AIME 2003, Protaras, Chypre*, Lecture Notes in Artificial Intelligence 2780, pages 304–313, Oct 2003.

[12] J. Lieber and A. Napoli. Correct and Complete Retrieval for Case-Based Problem-Solving. In H. Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98), Brighton, United Kingdom*, pages 68–72, 1998.

[13] E. Melis, J. Lieber, and A. Napoli. Reformulation in Case-Based Reasoning. In B. Smyth and P. Cunningham, editors, *Fourth European Workshop on Case-Based Reasoning, EWCBR-98*, Lecture Notes in Artificial Intelligence 1488, pages 172–183. Springer, 1998.

[14] A. Napoli, C. Laurenço, and R. Ducournau. An object-based representation system for organic synthesis planning. *International Journal of Human-Computer Studies*, 41(1/2):5–32, 1994.

[15] B. Nebel. *Reasoning and Revision in Hybrid Representation Systems*. Lecture Notes in Computer Science 422. Springer-Verlag, Berlin, 1990.

[16] N. Noy, R. Fergerson, and M. Musen. The knowledge model of Protégé-2000: Combining interoperability and flexibility. In *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management, EKAW 2000*, October 2000.

[17] N. F. Noy and M. A. Musen. PROMPTDIFF: A Fixed-Point Algorithm for Comparing Ontology Versions. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence (AAAI/IAAI-02)*, pages 744–750, Menlo Parc, CA, USA, 2002. AAAI Press.

[18] C. K. Riesbeck and R. C. Schank. *Inside Case-Based Reasoning*. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, 1989.

[19] M. Rios, E. Desandes, B. Bresson, I. Klein, A. Lesur, F. Boisson, V. Demange, and P. Bey. Référentiels de bonnes pratiques cancérologiques : étude comparative de trois supports d'aide à la proposition thérapeutique pour les cancers du sein et de la prostate en Lorraine. *Bulletin du cancer*, 90(4), 2003.

[20] E. H. Ruspini, P. P. Bonissone, and W. Pedrycz, editors. *Handbook of Fuzzy Computation*. Institute of Physics Publishing, 1998.

[21] C. Sauvagnac. *La construction de connaissances par l'utilisation et la conception de procédures. Contribution au cadre théorique des activités métafonctionnelles*. Thèse d'Université, Conservatoire National des Arts et Métiers, 2000.

[22] C. Sauvagnac, J. Stines, A. Lesur, P. Falzon, and P. Bey. Application of therapeutic protocols: a tool to manage medical knowledge. In W. Horn, Y. Shahar, G. Lindberg, S. Andreassen, and J. Wyatt, editors, *Proceedings of the Joint European Conference on Artificial Intellingence in Medicine and Medical Decision Making (AIMDM-99)*, volume 1620 of *LNAI*, Berlin, 1999. Springer.

[23] U. Straccia. Reasoning within Fuzzy Description Logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.

[24] A. Wald. *Statistical Decision Functions*. Wiley, New York, 1950.

[25] Web-Ontology (WebOnt) Working Group. Web Ontology Language (OWL) Reference Version 1.0. W3C Candidate Recommendation, August 2003. `http://www.w3.org/TR/owl-ref`.