

Failure Analysis for Domain Knowledge Acquisition in a Knowledge-Intensive CBR System

Amélie Cordier¹, Béatrice Fuchs¹, Jean Lieber², and Alain Mille¹

¹ LIRIS CNRS, UMR 5202, Université Lyon 1, INSA Lyon, Université Lyon 2, ECL
43, bd du 11 Novembre 1918, Villeurbanne Cedex, France

{Amelie.Cordier,Beatrice.Fuchs,Alain.Mille}@liris.cnrs.fr

² Orpailleur team, LORIA UMR 7503 CNRS, INRIA, Nancy Universities
BP 239 54 506 Vandœuvre-lès-Nancy, France

Jean.Lieber@loria.fr

Abstract. A knowledge-intensive case-based reasoning system has profit of the domain knowledge, together with the case base. Therefore, acquiring new pieces of domain knowledge should improve the accuracy of such a system. This paper presents an approach for knowledge acquisition based on some failures of the system. The CBR system is assumed to produce solutions that are consistent with the domain knowledge but that may be inconsistent with the expert knowledge, and this inconsistency constitutes a failure. Thanks to an interactive analysis of this failure, some knowledge is acquired that contributes to fill the gap from the system knowledge to the expert knowledge. Another type of failures occurs when the solution produced by the system is only partial: some additional pieces of information are required to use it. Once again, an interaction with the expert involves the acquisition of new knowledge. This approach has been implemented in a prototype, called FRAKAS, and tested in the application domain of breast cancer treatment decision support.

1 Introduction

A case-based reasoning system (CBR [17]) relies on several containers of knowledge. The source cases are, obviously, among those containers of knowledge, but a lot of systems also use additional knowledge sources as the “domain knowledge” (also known as “domain ontology” or “domain theory”). The more correct and accurate the domain knowledge is, the better the CBR system’s inferences will be.

This paper presents an approach to interactive acquisition of domain knowledge in a CBR system. More precisely, this acquisition is performed during a CBR session: the target problem is automatically solved by adaptation of the retrieved case and, after that, the solution is presented to the user who, depending on his/her expertise level, may be able to detect that the solution is not satisfactory and why that is not the case. Two kinds of failures are considered in this paper:

- (1) The suggested solution is inconsistent with the expert knowledge and
- (2) The suggested solution is only partially valid (the user misses some information to fully exploit it).

An interactive mechanism that aims at incorporating new pieces of domain knowledge is described. The new knowledge is used to repair the failed adaptation and to prevent similar failures to occur in future reasonings. This work concerns domain knowledge acquisition during the CBR step called repairing in [17], also known as the revise step in [2]. Thus, retrieval and adaptation issues are not detailed in this paper.

The rest of the paper is organized as follows. In section 2, we present the notions, notations, and assumptions we make about CBR. Then the principles of our knowledge acquisition approach are outlined (section 3). Those principles have been implemented in a prototype called FRAKAS whose presentation in section 4 constitutes the core of the paper. An example and an algorithm show how FRAKAS assists the proposed acquisition method. Section 5 presents some related work on domain knowledge acquisition in CBR systems in comparison with the approach introduced in this paper. Finally, section 6 concludes this paper and proposes some future work.

2 Basic Notions, Notations and Assumptions on CBR

In this work, the notions of problem and solution are assumed to be well-defined. If pb is a problem (resp., sol is a solution) then pb (resp., sol) is an expression in a knowledge representation formalism representing a problem (resp., a solution) of this domain. In addition, it is assumed that there exists a binary relation that links a solution sol to a problem pb and meaning “ sol is a solution of pb ”. In some CBR applications, this relation is only imperfectly specified. However, a finite set of pairs $(srce, Sol(srce))$ is still available, where $srce$ is a problem and $Sol(srce)$ is a solution of $srce$. This finite set is the *case base* and a pair $(srce, Sol(srce))$ is a *source case*. We also denote by DK the knowledge base containing the domain knowledge.

Reasoning from cases means solving a problem called the target problem and denoted by tgt , using the case base. This reasoning process is usually constituted of two main steps: retrieval that aims at selecting a source case deemed to be similar to the target problem and adaptation that aims at solving the target problem by using the retrieved source case. Moreover, a third step is sometimes added, namely the learning step, that can be performed automatically or in interaction with an expert. This step consists in improving the system’s knowledge (cases, domain knowledge, etc.) after the adaptation of the retrieved case.

In this paper, we make four additional assumptions that we believe relevant for a CBR system. The first assumption is that the adaptation produces a result *consistent* with the domain knowledge (but not necessarily with the knowledge of the expert).

The second assumption is that there exists a computable distinction between a solution that totally solves a problem and a solution that only partially solves it. A way to distinguish between partial and non partial solutions is to split the vocabulary for representing cases in two subsets: the “abstract” vocabulary and “concrete”. If a solution needs some of the abstract vocabulary in order to be represented, then this solution is said to be partial.

The third assumption is that each problem (resp., solution) coded in the CBR system represents a set of problem instances (resp., a set of solution instances).

The fourth assumption is that a problem (resp., a solution) is represented by a set of descriptors interpreted in a conjunctive way: if $pb = \{d_1, \dots, d_n\}$, then pb describes the problem whose instances satisfy each of the descriptors d_i ($i \in \{1, \dots, n\}$). This assumption is not mandatory in the approach presented in this paper but makes simpler its explanation.

The example used in this paper addresses a specific domain, namely breast cancer treatment.

3 Principles

Between the domain knowledge DK in the CBR system and the expert's knowledge, there is usually a gap. According to [15], it is impossible to fully fill this gap in most of the practical applications: this is the so-called qualification problem. Nevertheless, new knowledge can still be acquired from the expert.

The general principle of the approach described here is to perform an on-line knowledge acquisition by analyzing the adaptation failures. An on-line acquisition is performed when the system is used: the system interacts with the expert to acquire some of his/her knowledge. One can talk of acquisition by failure analysis if the interaction with the expert relies on the fact that, according to the expert, the result is, at least partially, a failure. Two kinds of failures are considered in this paper. Each of them leads to specific knowledge acquisition (though quite similar). Other kinds of failures are likely to exist but are not considered here.

First kind of failures: inconsistency of the adapted solution with the expert knowledge. The expert points out that, considering his or her domain knowledge, the assessment “ $Sol(tgt)$ solves tgt ” is inconsistent. This can mean that the solution by itself is inconsistent (or unrealizable, such as the fact of transforming a cooked egg into a fresh egg) or that the solution is inconsistent with the context of the target problem (for example, if the problem tgt is “How to travel from Lyon to Belfast?” and its solution $Sol(tgt)$ is a plan to travel from Nancy to Aberdeen).

In both situations, the expert is supposed to highlight (thanks to an appropriate interface) a part of $Sol(tgt)$ (ideally, the “smallest” possible) that is inconsistent with his/her knowledge about the target problem. This part of the solution is a subset Inc of the set of descriptors of $Sol(tgt)$. A first acquired knowledge (added to DK) is the fact that “ Inc is false”. Then, the CBR process is performed again, with the new domain knowledge.

Afterwards, the expert is required for an explanation. This explanation may be complex and our opinion is that it is very complicated (if not impossible) to completely automate this part (modeling and formalizing knowledge is the matter of knowledge engineers and requires competences that a domain expert may not have). Therefore, the expert is invited to write an explanation in plain text. The resulting document is used later by a knowledge engineer, in presence of the expert, to acquire new knowledge (that will imply, in particular, but not only, that “ Inc is false”).

Second kind of failures: failures caused by a partial solution. If the solution $Sol(tgt)$ proposed after the adaptation is partial, and therefore, not fully satisfactory,

the interaction with the expert may make it precise. Let SI be the set of the instances of $\text{Sol}(\text{tgt})$. If such an instance $s \in SI$ is judged satisfactory by the expert, it constitutes a solution to the problem tgt . If, by contrast, it is inconsistent with the expert knowledge, the expert is supposed to highlight a minimal part Inc of the descriptors of s . Then, this amounts to the same knowledge acquisition process as the one proposed for the first kind of failures: “ Inc is false” is added to DK and the expert is asked for an explanation to latter support the knowledge acquisition with the expert and the knowledge engineer.

4 FRAKAS: A System for Domain Knowledge Acquisition by Interactive Analysis of Reasoning Failures in Case-Based Reasoning

FRAKAS (*FailuRe Analysis for domain Knowledge AcquiSition*) is a prototype that implements the principles introduced above with a knowledge representation in propositional logic.

4.1 Principles of the Adaptation

Formalism. The formalism used is propositional logic on a set of variables \mathcal{V} . Thus, DK , srce , $\text{Sol}(\text{srce})$, and tgt are propositional formulas on \mathcal{V} . \mathcal{V} is partitioned in $\{\mathcal{V}_{\text{pb}}^c, \mathcal{V}_{\text{pb}}^a, \mathcal{V}_{\text{sol}}^c, \mathcal{V}_{\text{sol}}^a, \mathcal{V}_{\text{other}}\}$ where $\mathcal{V}_{\text{pb}}^x$ (resp., $\mathcal{V}_{\text{sol}}^x$) represents the variables used to represent some problems (resp., some solutions), and if $x = c$ (resp., $x = a$) then this set only contains variables said to be concrete (resp., abstract). A problem (resp., a solution) is a formula whose variables belongs to $\mathcal{V}_{\text{pb}}^c \cup \mathcal{V}_{\text{pb}}^a$ (resp., to $\mathcal{V}_{\text{sol}}^c \cup \mathcal{V}_{\text{sol}}^a$). This distinction between problem variables and solution variables allows one to express a source case as a conjunction of its problem part and its solution part: $\text{srce-case} = \text{srce} \wedge \text{Sol}(\text{srce})$.

An interpretation on \mathcal{V} is a function \mathcal{I} that to $x \in \mathcal{V}$ associates $x^{\mathcal{I}} \in \{\text{T}, \text{F}\}$. \mathcal{I} is prolonged on the set of the formulas build on \mathcal{V} in the usual way (for example, $(f \wedge g)^{\mathcal{I}} = \text{T}$ iff $f^{\mathcal{I}} = \text{T}$ and $g^{\mathcal{I}} = \text{T}$). \mathcal{I} is a model of f if $f^{\mathcal{I}} = \text{T}$. f implies g (resp., f is equivalent to g), noted $f \models g$ (resp. $f \equiv g$) if $\text{Mod}(f) \subseteq \text{Mod}(g)$ (resp., $\text{Mod}(f) = \text{Mod}(g)$). f implies (resp., is equivalent to) g modulo DK , noted $f \models_{DK} g$ (resp., by $f \equiv_{DK} g$) if $DK \wedge f \models g$ (resp., if $DK \wedge f \equiv DK \wedge g$). $\text{Mod}(f)$ denotes the set of the models of f . The instances of a problem (resp., a solution) are defined here as its interpretations on $\mathcal{V}_{\text{pb}}^c \cup \mathcal{V}_{\text{pb}}^a$ (resp., on $\mathcal{V}_{\text{sol}}^c \cup \mathcal{V}_{\text{sol}}^a$).

A solution sol is partial if it is not possible to express it without any abstract variable, in other word, if there exists no f such that $\text{sol} \equiv_{DK} f$ and such that no variable of f belongs to $\mathcal{V}_{\text{sol}}^a$ ¹.

Conservative adaptation. The adaptation performed by FRAKAS follows the principle of conservative adaptation that is briefly described here (see [14] for more details).

¹ This is checked in FRAKAS as follows. For each $\mathcal{I} \in \text{Mod}(\text{sol})$, let \mathcal{I}^- be the interpretation obtained by projection of \mathcal{I} on the set of variables $\mathcal{V} \setminus \mathcal{V}_{\text{sol}}^a$. Then, let sol^- be a formula whose models are the \mathcal{I}^- 's, for $\mathcal{I} \in \text{Mod}(\text{sol})$. Then the test $\text{sol} \equiv_{DK} \text{sol}^-$ is done; sol can be written without any abstract solution variable (i.e., sol is not partial) iff this test holds.

This approach to adaptation consists in doing minimal changes on the source case in order to be coherent with both the target problem and the domain knowledge. This adaptation is formalized based on the notion of revision operator [11]: a revision operator \circ associates to two knowledge bases ψ and μ that entails μ and the knowledge base $\psi \circ \mu$ which, intuitively, is obtained by a minimal change on ψ to be consistent with μ . In the framework of propositional logic, the conservative adaptation of a case *srce-case* to solve a problem *tgt*, given the domain knowledge *DK* and a revision operator \circ is:

$$CA_{\circ}(DK, \text{srce-case}, \text{tgt}) = (DK \wedge \text{srce-case}) \circ (DK \wedge \text{tgt})$$

A solution $\text{Sol}(\text{tgt})$ can be deductively inferred. From a practical viewpoint, Dalal’s revision operator, noted \circ_D [11], is used. \circ_D is defined as follows. Let dist be the Hamming distance between interpretations on \mathcal{V} ($\text{dist}(\mathcal{I}, \mathcal{J})$ is the number of $x \in \mathcal{V}$ such as $x^{\mathcal{I}} \neq x^{\mathcal{J}}$) and let $G^\lambda(\psi)$ be the formula (for $\lambda \geq 0$ and ψ a formula) such that:

$$\text{Mod}(G^\lambda(\psi)) = \{ \mathcal{J} \mid \mathcal{J} : \text{interpretation on } \mathcal{V} \text{ such as exists} \\ \mathcal{I} \in \text{Mod}(\psi) \text{ with } \text{dist}(\mathcal{I}, \mathcal{J}) \leq \lambda \}$$

(This defines $G^\lambda(\psi)$ up to the logical equivalence, which is enough since we adhere to the principle of irrelevance of syntax, saying that whenever $f \equiv g$, an artificial reasoning system using knowledge f makes the same inferences –up to logical equivalence– as the same system using g instead of f .) For ψ and μ two formulas such that at least the latter is satisfiable, $\psi \circ_D \mu$ is defined as being $G^\Delta(\psi) \wedge \mu$ where Δ is the smallest value such as $G^\Delta(\psi) \wedge \mu$ is satisfiable. Intuitively, $\psi \circ_D \mu$ is obtained by generalizing ψ minimally (according to the scale $(\{G^\lambda\}_\lambda, \models)$) to be consistent with μ .

Example. Léon is about to invite Thècle and wants to prepare her an appropriate meal. His target problem can be specified by the characteristics of Thècle about food. Let us assume that Thècle is vegetarian (denoted by the propositional variable v) and that she has other characteristics (denoted by o) not detailed in this example: $\text{tgt} = v \wedge o$. From his experience as a host, Léon remembers that he had invited Simone some time ago and he thinks that Simone is very similar to Thècle according to food preferences, except that she is not a vegetarian: $\text{srce} = \neg v \wedge o$. He had proposed to Simone a meal with salad (s), beef (b) and a dessert (d), and she was satisfied by the two formers but has not eaten the dessert. Thus Léon has retained the case $(\text{srce}, \text{Sol}(\text{srce}))$ with $\text{Sol}(\text{srce}) = s \wedge b \wedge \neg d$. Besides that, Léon has some general knowledge about food: he knows that beef is meat, that meat and tofu are protein-based food, that tofu is not meat, and that vegetarians do not eat meat. Thus, his domain knowledge is

$$DK = b \rightarrow m \quad \wedge \quad m \rightarrow p \quad \wedge \quad t \rightarrow p \quad \wedge \quad \neg t \vee \neg m \quad \wedge \quad v \rightarrow \neg m$$

On this example, conservative adaptation produces the following result:

$$CA_{\circ_D}(DK, \text{srce-case}, \text{tgt}) \equiv_{DK} \underbrace{v \wedge o}_{\text{tgt}} \quad \wedge \quad \underbrace{s \wedge \neg m \wedge p \wedge \neg d}_{\text{Sol}(\text{tgt})}$$

if Léon follows $\text{Sol}(\text{tgt})$, he will propose to Thècle a dinner with a salad, a main course with proteins but no meat (for example, a tofu-based dish) and no dessert.

4.2 Study of an Example Through FRAKAS

The example described in details in this section comes from the research project KASIMIR whose framework is knowledge management and decision support in oncology [8]². A problem is given by the description of a patient suffering from breast cancer. A solution is a therapy. In this example, the successive states of the domain knowledge are denoted by DK_0, DK_1 , etc.

Example specification. Jules is a man suffering from breast cancer with other characteristics not detailed here (in particular, the fact that the decision is made after the surgery that has removed the tumor, and the fact that the hormone receptors are positive). It can be noticed that this example comes from a real example, that has been simplified and for which the first name of the patient has been changed. It is modeled by the problem $\text{tgt} = \text{man} \wedge \text{other-charac}$.

If M_1 and M_2 are two sets of interpretations, $\text{dist}(M_1, M_2)$ denotes the minimum of the values $\text{dist}(\mathcal{I}_1, \mathcal{I}_2)$, for $\mathcal{I}_1 \in M_1$ and $\mathcal{I}_2 \in M_2$. As argued in [14], the following criterion for retrieval (following the principle of adaptation-guided retrieval [19], for the conservative adaptation based on \circ_D) can be given: the source case srce-case^1 has to be preferred to the source case srce-case^2 when $\Delta^1 < \Delta^2$, with $\Delta^i = \text{dist}(\text{Mod}(\text{srce-case}^i), \text{Mod}(\text{tgt}))$ ($i \in \{1, 2\}$).

It is assumed that exists a source case $\text{srce-case} = (\text{srce}, \text{Sol}(\text{srce}))$ such that $\text{srce} = \text{woman} \wedge \text{other-charac}$. This source case corresponds to a woman having the same characteristics as Jules, except for her gender. This source case has been retrieved because it is very similar to tgt according to a (conservative) adaptation-guided retrieval criterion defined in [14].

The solution of this problem is $\text{Sol}(\text{srce}) = \text{FEC-50} \wedge \text{Rad-50Gy} \wedge \text{ovariectomy}$: this treatment corresponds to a cure of FEC 50 (a chemotherapy drug), a breast radiotherapy with a dose of 50 Gy, and an ovariectomy (ovary ablation), that has an anti-oestrogen effect and, so, constitutes a hormone therapy. There are other anti-oestrogen treatments, such as the treatment with tamoxifen and the one with anti-aromatases. The knowledge presented above, together with the fact that men are not women can be formalized by

$$\begin{aligned} DK_0 = & (\neg\text{woman} \vee \neg\text{man}) \wedge (\text{FEC-50} \rightarrow \text{chemotherapy}) \wedge \\ & (\text{Rad-50Gy} \rightarrow \text{radiotherapy}) \wedge (\text{ovariectomy} \rightarrow \text{anti-oestro}) \wedge \\ & (\text{tamoxifen} \rightarrow \text{anti-oestro}) \wedge (\text{anti-aromatases} \rightarrow \text{anti-oestro}) \wedge \\ & (\text{anti-oestro} \rightarrow \text{hormone-therapy}) \end{aligned}$$

Moreover, it is assumed that:

$$\begin{aligned} \mathcal{V}_{\text{sol}}^c = & \{\text{FEC-50}, \text{Rad-50Gy}, \text{ovariectomy}, \text{tamoxifen}, \text{anti-aromatases}\} \\ \mathcal{V}_{\text{sol}}^a = & \{\text{chemotherapy}, \text{radiotherapy}, \text{anti-oestro}, \text{hormone-therapy}\} \end{aligned}$$

² The medical knowledge presented here has been simplified and should not be considered correct from a medical viewpoint.

The conservative adaptation gives:

$$\begin{aligned} CA_{o_D}(DK_0, srce\text{-}case, tgt) &= (DK_0 \wedge srce \wedge Sol(srce)) \circ_D (DK_0 \wedge tgt) \\ &\equiv_{DK_0} man \wedge other\text{-}charac \wedge FEC\text{-}50 \wedge Rad\text{-}50Gy \wedge ovariectomy \end{aligned}$$

Knowledge acquisition following a detection, by the expert, of inconsistency of the solution with his/her knowledge. The result of the conservative adaptation is presented to the expert (figure 1). This latter is in charge of determining if the solution is consistent with his/her knowledge. In this example, this is not the case (type 1 failures) and he/she checks a set of literals such that their conjunction is inconsistent with his/her knowledge. He checks *man* and *ovariectomy* since he/she knows that it is not possible to do an ovariectomy on men (cf. figure 1). Therefore, $Inc_1 = man \wedge ovariectomy$ is false and $\neg Inc_1$ can be added to the domain knowledge:

$$DK_1 = DK_0 \wedge \neg Inc_1 \equiv DK_0 \wedge (\neg man \vee \neg ovariectomy)$$

Moreover, the expert is asked to provide an explanation, and he/she proposes the following one:

Text 1: *To make an ablation of ovaries on a person, it is necessary that this person has ovaries, which is not the case for men.*

Then, the system performs a new adaptation:

$$\begin{aligned} CA_{o_D}(DK_1, srce\text{-}case, tgt) &\equiv_{DK_1} man \wedge other\text{-}charac \wedge FEC\text{-}50 \wedge Rad\text{-}50Gy \\ &\quad \wedge \neg ovariectomy \wedge anti\text{-}oestro \end{aligned}$$

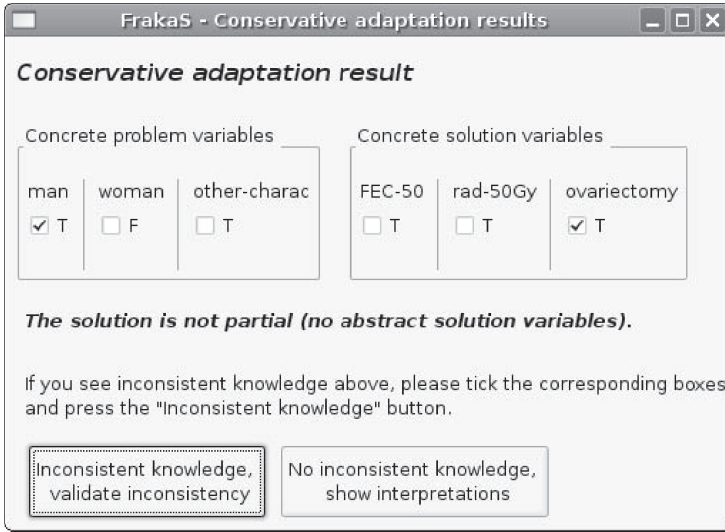
(the conservative adaptation does not keep the ovariectomy, since it is in contradiction with $DK_1 \wedge man$ but keeps the idea of an anti-oestrogen treatment).

Knowledge acquisition for making a partial solution precise. Then, the result of the second adaptation is presented to the expert. This latter, first indicates that the solution is consistent with his/her knowledge. Thus, there is no first type failure, but there is a second type failure: the type of anti-oestrogen treatment should be precised. Indeed, $anti\text{-}oestro \in \mathcal{V}_{so1}^a$ and there exists no formula f that does not contain any variable of \mathcal{V}_{so1}^a that is equivalent to $CA_{o_D}(DK_1, srce\text{-}case, tgt)$ modulo DK . In this situation, the set of interpretations of $CA_{o_D}(DK_1, srce\text{-}case, tgt)$ is presented to the expert who points out the ones that are inconsistent with his/her knowledge (cf. figure 2) and, for each of them, a set of literals whose conjunction Inc is inconsistent with his/her knowledge. In this example, 2 of the 4 interpretations (the first and the fourth on the figure) are inconsistent. From the first one, the expert makes a selection that corresponds to

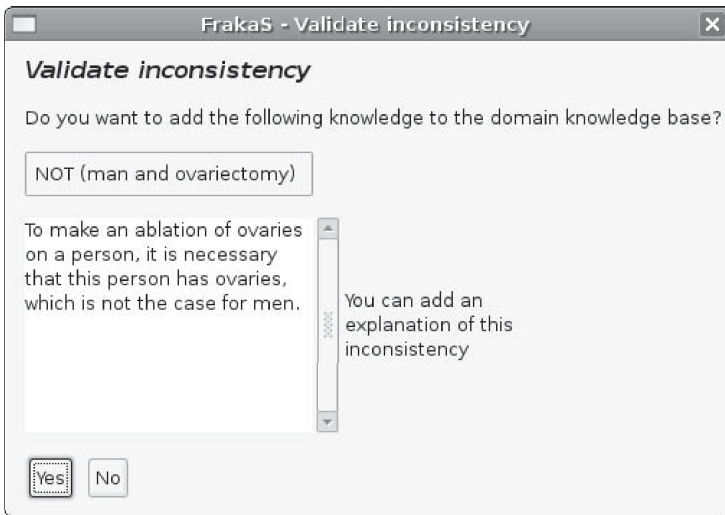
$$Inc_{2.1} = \neg ovariectomy \wedge \neg tamoxifen \wedge \neg anti\text{-}aromatases \wedge anti\text{-}oestro$$

And he/she explains it by text 2 and FRAKAS adds $\neg Inc_{2.1}$ to the domain knowledge:

Text 2: *The only therapies that are possible and permitted in my hospital for an anti-oestrogen treatment are the ovariectomy, the tamoxifen, and the anti-aromatases.*



(a) Display of $CA_{o_D}(DK_0, srce\text{-}case, tgt)$



(b) The expert validates the acquired knowledge and provides an explanation

Fig. 1. First solution presented to the expert ($CA_{o_D}(DK_0, srce\text{-}case, tgt)$) and his/her feedback, in the form of checked boxes (a). Plain text explanation provided by the expert (b).

$$\begin{aligned}
 DK_2 &= DK_1 \wedge \neg Inc_{2.1} \\
 &\equiv DK_1 \wedge \text{anti-oestro} \rightarrow \left(\begin{array}{c} \text{ovariectomy} \vee \text{tamoxifen} \\ \vee \text{anti-aromatases} \end{array} \right)
 \end{aligned}$$

For the other interpretation that is inconsistent with the expert knowledge, Inc is $Inc_{2.2} = \text{tamoxifen} \wedge \text{anti-aromatases}$ that is explained by text 3.

Fraka5 - Conservative adaptation results

Conservative adaptation result

Concrete problem variables			Concrete solution variables			Abstract solution variables
man	woman	other-charac	FEC-50	rad-50Gy	ovariectomy	anti-oestrogens
<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T

The solution is partial (anti-oestrogens).

if you see inconsistent knowledge above, please tick the corresponding boxes and press the "Inconsistent knowledge" button.

(a) Display of $CA_{op}(DK_1, srce\text{-}case, tgt)$ that is judged by the expert to be consistent

Fraka5 - Conservative adaptation results (interpretations)

Conservative adaptation result (interpretations)

Concrete problem variables			Concrete solution variables			Abstract solution variables	Concrete problem variables	
man	woman	other-charac	FEC-50	rad-50Gy	ovariectomy	anti-oestrogens	anti-aromatases	tamoxifen
<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> T	<input checked="" type="checkbox"/> F	<input checked="" type="checkbox"/> T	<input checked="" type="checkbox"/> F	<input checked="" type="checkbox"/> F
<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> F
<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T
<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> T	<input type="checkbox"/> F	<input type="checkbox"/> T	<input checked="" type="checkbox"/> T	<input checked="" type="checkbox"/> T

The solution is not partial (no abstract solution variables).

if you see on or more inconsistent interpretations above, please tick the corresponding boxes and press the "Inconsistent interpretations" button.

(b) Display of the interpretations of $CA_{op}(DK_1, srce\text{-}case, tgt)$ and the feedback of the expert on the interpretations he/she rejects

Fig. 2. Second solution presented to the expert and his/her feedback (only the checking box part)

Text 3: A given hormone therapy should not use at the same time tamoxifen and anti-aromatases.

$$DK_3 = DK_2 \wedge \neg Inc_{2,2} \equiv DK_2 \wedge (\neg tamoxifen \vee \neg anti\text{-}aromatases)$$

Then, $\neg Inc_{2,2}$ is added to DK_2 .

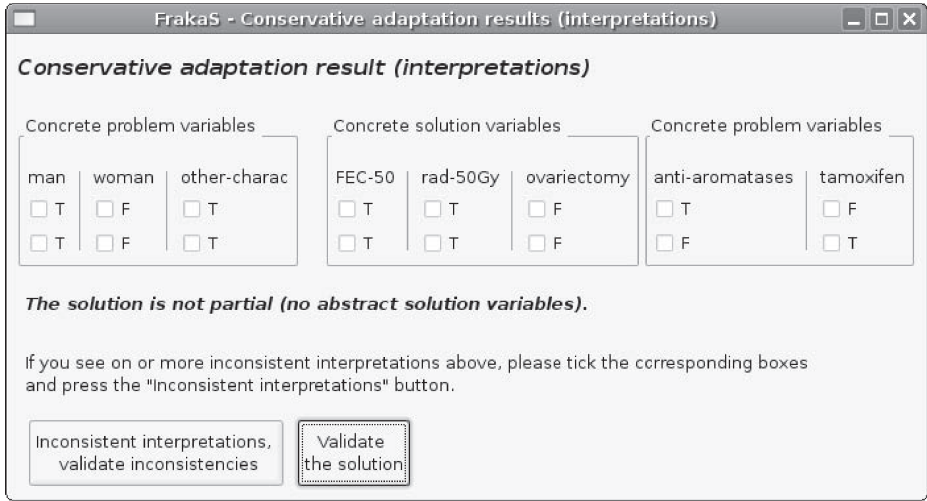


Fig. 3. Third solution presented to the expert, who validates it

Then, conservative adaptation gives (cf. figure 3):

$$CA_{o_D}(DK_3, srce\text{-}case, tgt) \equiv_{DK_3} man \wedge other\text{-}charac \wedge FEC\text{-}50 \wedge Rad\text{-}50Gy \wedge \neg ovariectomy \wedge (tamoxifen \oplus anti\text{-}aromatases)$$

(where \oplus is the symbol of exclusive or) that is validated by the expert (no type 1 failure) and can be written without using abstract variables of solution (no type 2 failure). This formula has two interpretations: the first one recommends tamoxifen and the second one, anti-aromatases.

Taking into account the explanations. The three texts given by the expert can be used as sources for acquiring some new domain knowledge to be added to $DK_3 = DK_0 \wedge \neg Inc_1 \wedge \neg Inc_{2,1} \wedge \neg Inc_{2,2}$. It can be noticed that this new knowledge acquisition (by contrast to the one presented above) is off-line; it is performed during knowledge maintenance operations of the CBR system.

Taking into account the first text. In this text, a knowledge engineer can establish the following knowledge, thanks to discussions with the expert:

- A man does not have ovaries ($f_1 = man \rightarrow \neg has\text{-}ovaries$);
- If a person has to be treated by ovariectomy, then this person must have ovaries ($f_2 = ovariectomy \rightarrow has\text{-}ovaries$);
- A woman who has already had an ovariectomy does not have her ovaries any more ($f_3 = antecedent\text{-}ovariectomy \rightarrow \neg has\text{-}ovaries$).

f_1 and f_2 formalize the text 1. f_3 comes from an answer of the expert to the following question asked by the knowledge engineer: “Are there women that do not have ovaries?”

Then, the state of the domain knowledge is:

$$DK_4 = DK_3 \wedge f_1 \wedge f_2 \wedge f_3$$

It can be noticed that during this phase, the vocabulary of the CBR system is enriched. It can also be noticed that $DK_0 \wedge f_1 \wedge f_2 \models DK_0 \wedge \text{Inc} = DK_1: f_1$ and f_2 explain Inc_1 that has to be a consequence of their conjunction. But, the additional knowledge f_3 enables to solve correctly the problem $\text{tgt}' = \text{woman} \wedge \text{antecedent-ovariectomy} \wedge \text{other-charac}$ by adaptation of the same source case:

$$\text{CA}_{\text{OD}}(DK_4, \text{srce-case}, \text{tgt}') \equiv_{DK_4} \text{woman} \wedge \text{antecedent-ovariectomy} \wedge \text{other-charac} \\ \wedge \text{FEC-50} \wedge \text{Rad-50Gy} \wedge \neg \text{ovariectomy} \wedge (\text{tamoxifen} \oplus \text{anti-aromatases})$$

Taking into account the second text. From the second text, the following fact can be acquired: when an anti-oestrogen treatment is required it is necessarily either an ovariectomy or a treatment with tamoxifen or anti-aromatases:

$$f_4 = \text{anti-oestro} \rightarrow (\text{ovariectomy} \vee \text{tamoxifen} \vee \text{anti-aromatases})$$

Nevertheless, this does not add new knowledge to what has already been acquired: $f_4 \equiv \neg \text{Inc}_{2.1}$. Does it imply that the second text is useless? No, since it highlights the fact that the knowledge f_4 (or $\neg \text{Inc}_{2.1}$) is *contextual*: it holds in the framework of the expert's hospital but a discussion with the expert points out that there exist other types of anti-oestrogen treatments. Therefore, it is important to avoid using this knowledge $\neg \text{Inc}_{2.1}$ in another medical context.

Taking into account the third text. A formalization of the third text gives $f_5 = \neg \text{tamoxifen} \vee \neg \text{anti-aromatases}$ but this does not enrich the domain knowledge: $f_5 \equiv \neg \text{Inc}_{2.2}$.

4.3 Main Algorithm of FRAKAS

Input: a problem tgt , a case base, and a domain knowledge DK

begin (algorithm)

The case retrieval with target problem tgt gives the source case $(\text{srce}, \text{Sol}(\text{srce}))$.

$\text{Sol}(\text{tgt}) \leftarrow \text{Adaptation}(DK, (\text{srce}, \text{Sol}(\text{srce})), \text{tgt})$

/ Taking into account type 1 failures */*

while the user finds that $\text{tgt} \wedge \text{Sol}(\text{tgt})$ is inconsistent with his/her knowledge

The user points out the failure Inc and gives a textual explanation.

$DK \leftarrow DK \wedge \neg \text{Inc}$

The textual explanation is stored for off-line knowledge acquisition.

$\text{Sol}(\text{tgt}) \leftarrow \text{Adaptation}(DK, (\text{srce}, \text{Sol}(\text{srce})), \text{tgt})$

end (while)

/ Taking into account type 2 failures */*

if $\text{Sol}(\text{tgt})$ is not partial **then** exit

while the user finds inconsistency is some interpretations of $\text{tgt} \wedge \text{Sol}(\text{tgt})$

```

for each inconsistent interpretation  $\mathcal{I}$ 
  The user points out the failure Inc and gives a textual explanation.
   $DK \leftarrow DK \wedge \neg \text{Inc}$ 
  The textual explanation is stored for off-line knowledge acquisition.
end (for)
 $\text{Sol}(\text{tgt}) \leftarrow \text{Adaptation}(DK, (\text{srce}, \text{Sol}(\text{srce})), \text{tgt})$ 
end (while)
end (algorithm)

```

5 Discussion and Related Work

By storing solved problems, CBR allows one to obtain solution hypothesis to new problems even in weak or incomplete theory domains. Meanwhile, these solutions may be non appropriate because of a lack of sufficient knowledge, leading to reasoning failures. Thus, many research work address the learning component in CBR systems which has been studied along several perspectives. One of these perspectives characterizes the different kinds of knowledge containers targeted by the learning process [16]: cases, similarity knowledge, adaptation knowledge and domain knowledge. Another perspective characterizes the knowledge source used by the learning process [21].

Some approaches use the content of the knowledge containers, in particular those who rely on machine-learning techniques in order to explicit hidden knowledge [10; 6; 7]. Other approaches, by contrast, aim at acquiring new knowledge that is not already in the system through interactions with the environment [4; 13]. FRAKAS can be classified in this second category. Learning takes place during the use of the system and aims at acquiring domain knowledge. The evaluation of the adapted solution may highlight the fact that it does not meet the requirements of the target problem. In this situation, a reasoning failure occurs that can be treated by a learning from failures process. The expert is involved in the process of identifying inconsistent parts of the solution whose negation constitutes new knowledge.

Among related approaches, the CHEF system [9], a case-based planner in the cooking domain, can be cited. CHEF uses a causal model to test an adapted plan. In case of failure, CHEF generates an explanation to guide the repair of the solution. Then, the learning process sets appropriate indexes in order to avoid a later retrieval of the faulty plan in similar circumstances. Besides case-based planning, CHEF inspired many subsequent lines of research based on explanations in order to search for failure causes, propose the associated repairs of the case solution, and modify the knowledge involved in the failure. Among work conducted on explanations, the METAQUA system [5] provides a taxonomy of failure causes associated to explanations in order to determine appropriate learning strategies. CREEK's reasoning and learning models [1] are built upon explanations in a knowledge-intensive context and [20] stresses the importance of explanations in the machine-learning process (and also for human learning and understanding).

In FRAKAS, textual explanations are used offline by knowledge engineers and domain experts to maintain domain knowledge. But FRAKAS also interacts with an expert during the reasoning process in a simple manner to point out faulty knowledge

and gives the opportunity to add a textual explanation. A parallel may be established between FRAKAS and the relevance feedback principle [18] of information retrieval where items are emphasized or weakened depending on user feedback. In relevance feedback, users are marking documents as relevant to their needs and this gives information to the information retrieval system on how to modify the query for better further retrievals. In FRAKAS, the user marks inconsistent knowledge which is integrated to domain knowledge and further adaptation is retried thanks to this modification. This kind of interaction is quite simple and intuitive for the user while it gives minimal but useful information to the system to enhance the process.

6 Conclusion and Future Work

This paper presents an approach for acquiring domain knowledge based on reasoning failures of a CBR system. This work is restricted to a framework where a system produces solutions that are consistent with its domain knowledge, as it is the case, for instance, when based on conservative adaptation. Two kinds of failures are considered. The first one is characterized by a conflict between the solution inferred by the CBR system and knowledge of the domain expert (though it is consistent with domain solutions). An analysis of this failure highlights the faulty descriptors which led to the conflict and then, add new domain knowledge. The second kind of failures is characterized by the fact that the solution is only partial: some information needed to make it usable is missing. If an analysis of solution instances shows that some of them are conflicting, the result of this analysis is used to acquire new knowledge. Furthermore, textual explanations provided by the expert constitute also a starting point for acquiring new knowledge or even to clarify the context of some knowledge pieces. This approach has been implemented in FRAKAS, a prototype based on propositional logic. This formalism has been chosen because it is a simple one for expressing inconsistencies, but the ideas presented here should be transposable to other formal frameworks (e.g. descriptions logics and fuzzy logics).

The work described in this paper is only in an early stage. Several further research directions may be considered. First, FRAKAS has to be improved to be usable in real-world situations: the core example of the paper is a use case of FRAKAS that we have designed (by simplifying a real medical situation). In a practical way, for the KASIMIR project, the system should be confronted to cancer specialists under the assistance of computer scientists. This entails to work on the interface ergonomics of FRAKAS and to the optimization of several parts of the code of FRAKAS. It would be interesting to study the opportunity of selecting relevant interpretations and relevant variables in order to reduce the complexity and to make the work of the expert easier.

Moreover, since the KASIMIR system is based on a description logic formalism (see [3]), it will be necessary either to implement translation procedures between propositional logic and description logic (these procedures are necessarily approximate) or to implement a new version of FRAKAS based on description logic.

We also need to go further into the failures of the second kind. Indeed, in our example, an interaction with the expert was sufficient to handle it but it may not be always the case. Suppose for example that there exists a great number of anti-oestrogen

treatments, ao_1, ao_2, \dots, ao_n , it will be tedious to enumerate them all and thus to obtain the knowledge $\text{anti-oestro} \rightarrow ao_1 \vee ao_2 \vee \dots \vee ao_n$. It seems more reasonable that the adaptation process provides such kind of result: “anti-oestrogen treatment, such as one based on tamoxifen or anti-aromatases”.

Finally, an underlying assumption of this work is that domain knowledge is at any time consistent with expert’s knowledge. This does not necessarily hold: DK may be “approximately true” (true in most situations but not all). In this case, when adding a new knowledge f to DK, a conflict of $DK \wedge f$ may occur. Consequently, such a conflict must be detected (which is not difficult). One could go further and propose to merge these two knowledge bases, by using a merging operator (see for example [12]). In particular, if one considers that DK can be revised, but f must be kept, one can use a revision operator instead of the ordinary conjunction: instead of $DK_{i+1} = DK_i \wedge f$, we would have $DK_{i+1} = DK_i \circ f$. This may occur if the use of FRAKAS leads first to the approximative knowledge $\neg \text{Inc}$ and then to the formula f which models the textual explanation given by the expert. This latter point needs to be studied thoroughly.

References

1. Aamodt, A.: A Knowledge-Intensive, Integrated Approach to Problem Solving and Sustained Learning. Doctoral dissertation, University of Trondheim, Norway (1991)
2. Aamodt, A., Plaza, E.: Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications* 7(1), 39–59 (1994)
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: *The Description Logic Handbook*. Cambridge University Press, Cambridge, UK (2003)
4. Cordier, A., Fuchs, B., Mille, A.: Engineering and Learning of Adaptation Knowledge in Case-Based Reasoning. In: Staab, S., Svátek, V. (eds.) *EKAW 2006*. LNCS (LNAI), vol. 4248, pp. 303–317. Springer, Heidelberg (2006)
5. Cox, M.T., Ram, A.: Introspective multistrategy learning: On the construction of learning strategies. *Artificial Intelligence* 112, 1–55 (1999)
6. Craw, S., Wiratunga, N., Rowe, R.C.: Learning adaptation knowledge to improve case-based reasoning. *Artificial Intelligence* 170(16–17), 1175–1192 (2006)
7. d’Aquin, M., Badra, F., Lafrogne, S., Lieber, J., Napoli, A., Szathmary, L.: Case Base Mining for Adaptation Knowledge Acquisition. In: *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI’07)*, pp. 750–755. Morgan Kaufmann, Inc, San Francisco (2007)
8. d’Aquin, M., Lieber, J., Napoli, A.: Adaptation Knowledge Acquisition: a Case Study for Case-Based Decision Support in Oncology. *Computational Intelligence (an International Journal)* 22(3/4), 161–176 (2006)
9. Hammond, K.J.: Explaining and Repairing Plans That Fail. *AI Magazine* 45(1–2), 173–228 (1990)
10. Hanney, K., Keane, M.T.: Learning Adaptation Rules From a Case-Base. In: Smith, I., Faltings, B. (eds.) *Advances in Case-Based Reasoning*. LNCS, vol. 1168, pp. 179–192. Springer, Berlin (1996)
11. Katsuno, H., Mendelzon, A.: Propositional knowledge base revision and minimal change. *Artificial Intelligence* 52(3), 263–294 (1991)
12. Konieczny, S., Lang, J., Marquis, P.: DA^2 merging operators. *Artificial Intelligence* 157(1–2), 49–79 (2004)

13. Leake, D.B., Kinley, A., Wilson, D.C.: Learning to Integrate Multiple Knowledge Sources for Case-Based Reasoning. In: Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97), pp. 246–251 (1997)
14. Lieber, J.: Application of the Revision Theory to Adaptation in Case-Based Reasoning: the Conservative Adaptation. In: Proceedings of the 7th International Conference on Case-Based Reasoning. LNCS, vol. 4626, pp. 224–238. Springer, Heidelberg (2007)
15. McCarthy, J.: Epistemological Problems of Artificial Intelligence. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence (IJCAI'77, Cambridge (Massachusetts), pp. 1038–1044 (1977)
16. Richter, M.M.: The Knowledge Contained in Similarity Measures. In: Aamodt, A., Veloso, M.M. (eds.) Case-Based Reasoning Research and Development. LNCS, vol. 1010, Springer, Heidelberg (1995)
17. Riesbeck, C.K., Schank, R.C.: Inside Case-Based Reasoning. Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey (1989)
18. Rocchio, J.J.: Document Retrieval Systems – Optimization and Evaluation. PhD thesis, Harvard University (March 1966)
19. Smyth, B., Keane, M.T.: Experiments On Adaptation-Guided Retrieval In Case-Based Design. In: Aamodt, A., Veloso, M. (eds.) Case-Based Reasoning Research and Development. LNCS, vol. 1010, pp. 313–324. Springer, Heidelberg (1995)
20. Sørmo, F., Cassens, J., Aamodt, A.: Explanation in Case-Based Reasoning –Perspectives and Goals. *Artificial Intelligence Review* 24(2), 109–143 (2005)
21. Wilke, W., Vollrath, I., Althoff, K.-D., Bergmann, R.: A Framework for Learning Adaptation Knowledge Based on Knowledge Light Approaches. In: Proceedings of the Fifth German Workshop on Case-Based Reasoning, pp. 235–242 (1997)