HAL open science

Opportunistic Acquisition of Adaptation Knowledge and Cases - The IakA Approach

Amélie Cordier, Béatrice Fuchs, Léonardo Lana de Carvalho, Jean Lieber, Alain Mille

To cite this version:

Amélie Cordier, Béatrice Fuchs, Léonardo Lana de Carvalho, Jean Lieber, Alain Mille. Opportunistic Acquisition of Adaptation Knowledge and Cases - The IakA Approach. European Conference on Case-Based Reasoning, Sep 2008, Trier, Germany. pp.150-164. hal-00364367

HAL Id: hal-00364367

https://hal.science/hal-00364367

Submitted on 25 Feb 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Opportunistic Acquisition of Adaptation Knowledge and Cases — The IakA approach

Amélie Cordier[1], Béatrice Fuchs[1], Léonardo Lana de Carvalho[3], Jean Lieber[2], and Alain Mille[1]

[1]LIRIS CNRS, UMR 5202, Université Lyon 1, INSA Lyon, Université Lyon 2, ECL.
{Amelie.Cordier, Beatrice.Fuchs, Alain.Mille}@liris.cnrs.fr
[2]LORIA UMR 7503 CNRS, INRIA, Universités de Nancy
Jean.Lieber@loria.fr
[3]LEACM-Cris, Université Lyon 2, Institut de Sciences de l'Homme (ISH) LIESP, Université Lyon 1, INSA Lyon
Leonardo.LanaDeCarvalho@univ-lyon2.fr

Abstract. A case-based reasoning system relies on different knowledge containers, including cases and adaptation knowledge. The knowledge acquisition that aims at enriching these containers for the purpose of improving the accuracy of the CBR inference may take place during design, maintenance, and also on-line, during the use of the system. This paper describes IakA, an approach to on-line acquisition of cases and adaptation knowledge based on interactions with an oracle (a kind of "ideal expert"). IakA exploits failures of the CBR inference: when such a failure occurs, the system interacts with the oracle to repair the knowledge base. IakA-NF is a prototype for testing IakA in the domain of numerical functions with an automatic oracle. Two experiments show how IakA opportunistic knowledge acquisition improves the accuracy of the CBR system inferences. The paper also discusses the possible links between IakA and other knowledge acquisition approaches.

1 Introduction

Case-based reasoning exploits knowledge, such as domain knowledge and adaptation knowledge, to perform inferences on cases. The more complete and accurate the knowledge is, the better the inferences are. Hence, building efficient knowledge bases is of particular importance. The building of the knowledge base for a CBR system is often done beforehand, during the design phase. However, in order to make systems capable of evolving, the knowledge base has to evolve as well, thus additional knowledge acquisition has to be possible during the system use. In systems offering such a possibility, the acquired knowledge is reused in further reasoning sessions to improve the solutions produced.

Several ways of performing knowledge acquisition have been explored in CBR related research. Knowledge engineers and domain experts can collaborate to model knowledge of the domain. This manual approach is efficient because it

allows the acquisition of relevant knowledge coming from the expert but it is rather constrained by the availability of the expert and of the knowledge engineer. Other approaches rely on the knowledge already available in the system (often in the cases) to infer new knowledge, like adaptation rules. These approaches are efficient in the sense that they automate the acquisition process but they produce a large amount of knowledge that has to be validated by an expert. Moreover, this validation phase is performed off-line, out of a specific context, thus it may be felt by an expert as an irksome task. Hybrid approaches, such as IakA, combine the reasoning capabilities of the system with interactions with the expert to acquire missing knowledge in context.

Usually, knowledge acquisition approaches assume that the knowledge of the system is organized in separate knowledge containers and that the reasoning process is split into several distinct steps. These assumptions are helpful better to understand CBR, but they do not reflect the reality. Actually, knowledge containers are closely interconnected (not to say identical) and the steps of the CBR process contribute to the achievement of the same objective: problem solving. The adaptation-guided retrieval principle [11] is a good illustration of this point: adaptation knowledge is used to support retrieval, and retrieval and adaptation steps contribute to the problem solving. This paper advocates a unified view of CBR steps and knowledge containers. In IakA, the CBR process is considered as a whole and the knowledge acquisition process focuses on *the knowledge of the system*: cases and adaptation knowledge are acquired at the same time.

This paper presents IakA, an interactive and opportunistic approach for knowledge acquisition in CBR. IakA is interactive in so far as it exploits interactions between the expert and the system during CBR sessions. Its opportunistic aspect is due to the fact that reasoning failures trigger the acquisition process: the system seizes this opportunity to identify missing knowledge and to acquire it. One of the main advantages of this approach is that it focuses on knowledge known to be needed, which constitutes a strong guidance for the knowledge acquisition process and alleviates the effort required by the expert. In IakA, cases are adapted using adaptation knowledge. When a failure occurs, the applicability of the adaptation knowledge for this case has to be questioned. The expert plays two important roles: identifying the failure and correcting the faulty adaptation knowledge. As the adaptation knowledge is corrected in the context of the case being solved, it stays linked with the case. The case and its related adaptation knowledge are then added to the knowledge base.

The remainder of this paper is organized as follows. Section 2 compares several approaches of knowledge acquisition in CBR. Then, section 3 describes IakA, a set of principles for interactive knowledge acquisition in CBR systems which perform approximate reasoning (i.e when the aim is to find an approximate solution for a problem). The modelling of the expert by an *oracle* is discussed. Formalizing the adaptation knowledge acquisition process is described and the classical assumption of CBR—similar problems have similar solutions—is questioned. Section 4 is dedicated to IakA-NF, a prototypical CBR application implementing the principles of IakA in the numerical functions domain and

describes two experiments. Section 5 discusses the complementarity of the IAKA approach with other knowledge approaches introduced in section 2. Finally, section 6 concludes the paper and outlines some prospects for future work.

2 Knowledge acquisition in CBR

It has long been argued that CBR was a solution to the knowledge acquisition bottleneck in knowledge-based systems because it is easier to collect cases than other pieces of knowledge. However, CBR only partly overcomes this problem because it also requires substantial effort to acquire the knowledge involved in the reasoning process. As with other knowledge-based systems, the implementation of knowledge-intensive CBR systems has to cope with the knowledge acquisition problem, and this issue has motivated significant research.

Adaptation-guided retrieval [11] aims at retrieving a prior case that is the easiest to adapt, given the available adaptation knowledge. Thus the adaptation step is central and adaptation knowledge plays a major role in CBR. For this reason, several studies focus on adaptation knowledge acquisition to improve the global quality of the system [9, 2].

Knowledge acquisition takes place at different stages of the life cycle of a CBR system. Initial knowledge acquisition can be done with experts who manually model the domain knowledge, or with the assistance of automated learning methods. Such approaches are off-line in that they take place outside a CBR reasoning cycle. Among off-line methods, machine-learning techniques have been used for instance in [7]. In these methods, the case base is exploited to learn adaptation rules. Adaptation rules are generated by examining the differences between problems related to the differences between solutions. In the same vein, Craw et al. experiment further with this method by applying learning algorithms, such as C4.5, in the tablet formulation domain [3] . The CABAMAKA system uses a knowledge discovery process to acquire adaptation knowledge [4]. Data mining algorithms are applied to detect regularities which are candidates to become adaptation rules. Adaptation rules are then validated by a domain expert. Off-line methods have been successfully applied, nevertheless these methods do not make it possible to acquire knowledge that is not yet in the cases.

On-line methods take advantage of a reasoning cycle to learn from a problem-solving session. One of the first CBR systems, CHEF, a case-based planner in the cooking domain, experimented learning from failures [6]. CHEF learns by storing successfully adapted plans or repaired plans. When an adapted plan fails, CHEF builds a causal explanation of the failure in order to anticipate a future similar problem. Hammond qualifies his approach as an incremental repair process after a test or an execution: for a given problem, a first error-prone solution is produced and further tested and repaired incrementally using a causal model. CHEF differs in that it takes advantage of a failure to anticipate it in further reasoning cycles. DIAL is a disaster response planning system that retrieves and adapts prior similar past plans [10]. Adaptation is performed with help either of general adaptation rules, or of prior successful adaptation cases, or of the user.

Adaptation is a combination of transformations combined with memory search processes of knowledge required by the transformation. The adaptation effort is stored and reused for an adaptation-guided retrieval approach. FrakaS [1] is a system for enriching domain knowledge when failures due to the incompleteness of the knowledge base occur. When such a failure occurs, a knowledge acquisition process involving the domain expert is triggered. Interactions with the expert allow the system to add new knowledge to its knowledge base and to collect an explanation of the failure. This knowledge is stored and reused to avoid the failure reoccurring in further reasoning. FrakaS is an example of the opportunistic knowledge acquisition approach in which new knowledge is acquired from outside the system. Next section presents IakA, a complementary approach to FrakaS.

3 IakA: InterActive Knowledge Acquisition

IakA is an approach to interactive knowledge acquisition in CBR systems that produce approximate solutions. The main idea of the approach is to exploit reasoning failures and their repairs to acquire cases and adaptation knowledge. Indeed, the occurrence of a failure highlights the fact that knowledge is missing. When correcting a failure, the required knowledge is added to the knowledge base and is reused in the following reasoning sessions to improve the solutions. The acquisition process is made possible thanks to an *oracle* that is capable of correcting solutions and providing the necessary adaptation knowledge.

3.1 Definitions and hypotheses

In this work, the notions of *problem* and *solution* are assumed to be well defined. If $\mathtt{pb}$ is a problem (resp., $\mathtt{sol}$ is a solution), then $\mathtt{pb}$ (resp., $\mathtt{sol}$) is an expression in a knowledge representation formalism representing a problem (resp., a solution). $\mathcal{L}_{\mathtt{pb}}$ denotes the problem space and $\mathcal{L}_{\mathtt{sol}}$ denotes the solution space. Moreover, a binary relation on $\mathcal{L}_{\mathtt{pb}} \times \mathcal{L}_{\mathtt{sol}}$ is assumed to exist with the semantics "has for solution". This relation is generally not completely known by the system, but some of its instances are: they are the pairs $(\mathtt{srce}, \mathtt{Sol}(\mathtt{srce})) \in \mathcal{L}_{\mathtt{pb}} \times \mathcal{L}_{\mathtt{sol}}$, called *source cases*. The aim of the CBR process is to find a solution for the target problem denoted $\mathtt{tgt}$. $\widetilde{\mathtt{Sol}}(\mathtt{tgt})$ is a *candidate* solution of $\mathtt{tgt}$, i.e. the solution produced by the CBR system.

In order to adapt the solution of a case, IakA relies on *adaptation knowledge* mainly composed of *adaptation operators*.

Definition 1 (Adaptation operator —$\mathtt{AO}_r = (\mathtt{r}, \mathcal{A}_\mathtt{r})$).
An adaptation operator $\mathtt{AO}_r$ *is a pair* $(\mathtt{r}, \mathcal{A}_\mathtt{r})$ *where* $\mathtt{r}$ *is a binary relation between problems* ($\mathtt{r} \subseteq \mathcal{L}_{pb} \times \mathcal{L}_{pb}$). $\mathcal{A}_\mathtt{r}$ *is an adaptation function:*
if $(\mathtt{srce}, \mathtt{Sol}(\mathtt{srce}), \mathtt{tgt}) \in \mathcal{L}_{pb} \times \mathcal{L}_{sol} \times \mathcal{L}_{pb}$ *and* $\mathtt{srce}\ \mathtt{r}\ \mathtt{tgt}$
then $\mathcal{A}_\mathtt{r}(\mathtt{srce}, \mathtt{Sol}(\mathtt{srce}), \mathtt{tgt})$ *is a candidate solution of* $\mathtt{srce}$.

Adaptation operators are organized in adaptation methods. An adaptation method is linked to a source case.

Definition 2 (Adaptation method —AM(srce)).
The adaptation method $\mathit{AM}(srce)$ *associated with the case* $(\mathit{srce}, \mathit{Sol}(\mathit{srce}))$ *is a finite set of adaptation operators* $\mathit{AO}_r = (r, \mathcal{A}_r)$*. An adaptation method may also contain strategic knowledge for managing the adaptation operators.*

The notions of adaptation operators and adaptation methods can be likened respectively to adaptation specialists and adaptation strategies defined in [12]. The adaptation method is used to build a similarity path and an associated adaptation path.

Definition 3 (Similarity path —SP).
A similarity path from a problem srce *to a problem* tgt *is a set of* q *triples* $(\mathit{pb}_{i-1}, r_i, \mathit{pb}_i)$ *with :*

- pb_i *:* $q+1$ *problems;*
- $\mathit{pb}_0 = \mathit{srce}$ *and* $\mathit{pb}_q = \mathit{tgt}$*;*
- $\mathit{pb}_{i-1}\ r_i\ \mathit{pb}_i$ *(for* $i \in \{1, \ldots, q\}$*);*
- r_i *is such that* $(r_i, \mathcal{A}_{r_i})$ *is an available adaptation operator.*

$\mathcal{P}(\mathit{srce}, \mathit{tgt})$ *denotes the set of similarity paths that can be built from* srce *to* tgt*.*

The adaptation path is built after the similarity path.

Definition 4 (Adaptation path —AP).
The adaptation path AP *associated to a similarity path* SP *is a set of* q *triples* $(\widetilde{\mathit{Sol}}(\mathit{pb}_{i-1}), \mathcal{A}_{r_i}, \widetilde{\mathit{Sol}}(\mathit{pb}_i))$ *with :*

- $\widetilde{\mathit{Sol}}(\mathit{pb}_0) = \mathit{Sol}(\mathit{srce})$ *and* $\widetilde{\mathit{Sol}}(\mathit{pb}_q) = \widetilde{\mathit{Sol}}(\mathit{tgt})$*;*
- $\widetilde{\mathit{Sol}}(\mathit{pb}_i) = \mathcal{A}_{r_i}(\mathit{pb}_{i-1}, \widetilde{\mathit{Sol}}(\mathit{pb}_{i-1}), \mathit{pb}_i)$*.*

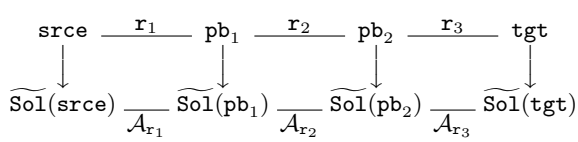

Fig. 1. A similarity path and the associated adaptation path.

Figure 1 shows an example of a similarity path and its adaptation path. In order to choose between several similarity paths relating the same srce and tgt, the notion of length of a similarity path is introduced. This notion relies on the notion of estimated adaptation error.

Definition 5 (Adaptation error —e_r and its estimation —$\widetilde{\mathrm{e}_r}$).
Each adaptation operator AO_r *introduces a numerical error* e_r*, function of the*

problems $\mathtt{srce}$ *and* $\mathtt{tgt}$ *related by* $\mathtt{r}$*:* $e_r(\mathtt{srce}, \mathtt{tgt}) \in \mathbb{R}_+$*. This error is known by the oracle but the system only knows an estimated value* $\widetilde{e_r}(\mathtt{srce}, \mathtt{tgt})$ *of it. Moreover,* $\widetilde{e_r}$ *is assumed to have the following property:* $\widetilde{e_r}(\mathtt{srce}, \mathtt{tgt}) = 0$ *if* $\mathtt{srce} = \mathtt{tgt}$*.*

Definition 6 (Length of a similarity path —$\ell(\mathtt{SP})$).

$$\ell(SP) = \sum_{i=1}^{q} \widetilde{e_r}(pb_{i-1}, pb_i)$$

Finally, the distance from a problem to another one is defined as the length of the shortest similarity path.[1]

Definition 7 (Distance between problems —$\mathtt{dist}(\mathtt{srce}, \mathtt{tgt})$).

$$dist(srce, tgt) = \min\{\ell(SP) \mid SP \in \mathcal{P}(srce, tgt)\}$$

Given these definitions, the retrieval process consists in building a similarity path, and consequently an adaptation path, from $\mathtt{srce}$ to $\mathtt{tgt}$ that minimizes the length $\ell(\mathtt{SP})$ and the adaptation process consists in following the adaptation path.

Illustration of the definitions. This example is given in a fictive domain where problems consist of ordered pairs of shapes and solutions consist of single shapes. Shapes have two properties: number of edges and color. It must be remarked that there is no available rule allowing the computation of the solution knowing the problem. Figure 2 illustrates the concept of adaptation operator. The candidate solution for the target problem (on the right) is obtained by adaptation of the source case (on the left). The relation $\mathtt{r}$ between $\mathtt{srce}$ and $\mathtt{tgt}$ means: *to go from* $\mathtt{srce}$ *to* $\mathtt{tgt}$*, a edge has to be added to the first shape of the pair.* Except this difference, all the relevant attributes of the shapes are identical. To $\mathtt{r}$ is associated the adaptation function $\mathcal{A}_{\mathtt{r}}$ which meaning is: *If there is one more edge on the first shape of the target problem, then the source solution must be adapted by adding one edge to it.* Hence, $\widetilde{\mathtt{Sol}}(\mathtt{tgt})$ is obtained by application of $\mathcal{A}_{\mathtt{r}}$ on $\mathtt{Sol}(\mathtt{srce})$.

3.2 Mechanisms of the IakA approach

The key idea of IakA is to exploit failures to acquire cases and adaptation knowledge. In systems that produce approximate solutions, a failure occurs when the distance between the solution of the system and the "ideal solution" is too large. IakA relies on the availability of an *oracle* which is able to say if a solution is satisfactory or not, to correct a non-satisfactory solution and to give adaptation

[1] Technically, an inf should be used instead of a min: it is possible to find a series of similarity paths $(\mathtt{SP}_n)_n$ such that $\ell(\mathtt{SP}_n) > 0$ and $\lim_{n\to\infty} \ell(\mathtt{SP}_n) = 0$. To avoid this theoretical problem, it is assumed that the number q of steps in a similarity path is bounded by a constant (e.g., $q \leq 10^{100}$).

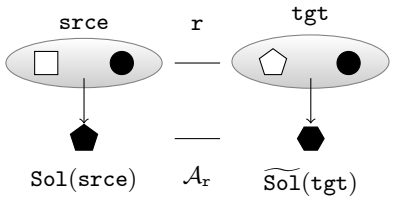

Fig. 2. An example of adaptation operator.

operators for a case. Hence, the oracle is able to compute a distance between solutions and to compare it to a *tolerance threshold* denoted by ε ($\varepsilon > 0$): if the distance is larger than ε, the solution is not satisfactory.

In CBR, the occurrence of a failure means that a piece of knowledge that was used during adaptation has to be corrected or made precise. In the framework of IAKA, adaptation methods, adaptation operators and adaptation errors may be questioned.

A main advantage of the IAKA approach is that the different pieces of knowledge (in particular, the adaptation operators) are separated and tested independently thus enabling the faulty knowledge to be identified more easily. Indeed, when a solution is not satisfactory, the adaptation operators involved are tested by the oracle one after the other. If the oracle identifies a faulty adaptation operator, it corrects it. The new piece of knowledge is added to the knowledge base of the system and a new CBR cycle is performed in order to find a better solution for the current problem. Adaptation operators are corrected and a new CBR cycle is performed until a satisfactory solution is found.

Justification of the IAKA approach. The CBR inference is based on the following principle (see, e.g., [5]):

Similar problems have similar solutions. (CBR principle)

The similarity between problems is the knowledge of the retrieval step, often in the form of a similarity measure or a distance between problems. The similarity between solutions is linked with the adaptation: the higher the error caused by adaptation is, the less the solutions are similar.

This principle can be replaced by its contraposition:

Dissimilar solutions solve dissimilar problems.

Therefore, a failure of the CBR inference indicates:

(a) Either that `srce` and `tgt` are not (enough) similar;
(b) Or a failure in the CBR principle.

The failure (a) can also be split into two sub-situations:

(a1) There is no source case similar to the target problem;
(a2) There is at least a source case $(\texttt{srce}', \texttt{Sol}(\texttt{srce}')) \neq (\texttt{srce}, \texttt{Sol}(\texttt{srce}))$ that is similar to $\texttt{tgt}$ but it has not been retrieved.

Each of the failures of type (a1), (a2), and (b) leads to a knowledge acquisition from the oracle.

When a failure of type (a1) occurs, the oracle may provide a new source case (with its associated adaptation method), that is similar to the target problem (for instance a case $(\texttt{tgt}, \texttt{Sol}(\texttt{tgt}))$ and an adaptation method $\texttt{AM}(\texttt{tgt})$).

When a failure of type (a2) occurs, this questions the similarity between problems that constitute the retrieval knowledge: $(\texttt{srce}, \texttt{Sol}(\texttt{srce}))$ is closer to $\texttt{tgt}$ than $(\texttt{srce}', \texttt{Sol}(\texttt{srce}'))$ and it should be the contrary. With a similarity based on the estimated adaptation errors, the interactions with the oracle should lead to a modification of these estimated errors.

When a failure of type (b) occurs, the similar problems $\texttt{srce}$ and $\texttt{tgt}$ have no similar solution. In other words, in a neighborhood of $\texttt{srce}$, the solution varies in an irregular manner. This situation can be interpreted with the notion of (dis)continuity of numerical functions $f : \mathbb{R}^n \to \mathbb{R}$. Indeed, if $\mathcal{L}_{\texttt{pb}} = \mathbb{R}^n$, $\mathcal{L}_{\texttt{sol}} = \mathbb{R}$, and $\texttt{Sol}(\texttt{pb})$ solves $\texttt{pb}$ if $f(\texttt{pb}) = \texttt{Sol}(\texttt{pb})$, then the continuity of f is defined intuitively with the CBR principle: if x_1 is close to x_2 then $f(x_1)$ is close to $f(x_2)$. A type (b) failure means that there is a discontinuity close to $\texttt{srce}$. The interactions with the oracle may be useful to better locate the discontinuity points. It may occur that these discontinuity points involve a partition of the problem space in several points. For example, if $\mathcal{L}_{\texttt{pb}} = \mathbb{R}$ and 4 is a discontinuity point highlighted by the oracle, then $\mathcal{L}_{\texttt{pb}}$ is partitioned in $\{]-\infty, 4[, \{4\},]4, +\infty[\}$. This implies that two problems of two different parts of this partition should never be considered as similar. With the previous example, 3.99 is dissimilar to 4.01. Therefore, the knowledge of this discontinuity point can be used as pieces of retrieval knowledge.

This justification of the IakA approach based on the CBR principle and the proximity of this principle to the notion of continuity suggest that it should be tested in domains where continuity is well-defined. The numerical functions constitute such domains. IakA-NF, described in the following section, is a prototype implementing IakA with numerical functions.

4 IakA-NF: a prototype implementing the IakA approach

4.1 The IakA-NF system

IakA-NF is a prototypical CBR engine implementing the principles of IakA in the application domain of the numerical functions ($f : \mathbb{R}^n \to \mathbb{R}$). The aim of this prototype is to solve problems by approximation, i.e., given n variables $(x_1, ..., x_n)$, the goal is to find an approximate value of $f(x_1, ..., x_n)$ by CBR.

In IakA-NF, a problem is a n-tuple of real numbers and a solution is an approximation of the value of the function f for these values. $f_{\texttt{CB}}$ denotes the

approximation of the function f obtained from the CBR system using the case base $\mathtt{CB}$ (and the adaptation methods). To each case is associated an adaptation method containing n adaptation operators. In the numerical functions domain, an intuitive way to define adaptation operators is to use the notion of partial derivatives. Indeed, the influence of the variation of a problem variable on the solution can be expressed by the partial derivative of this variable.

The retrieval is performed according to the distance defined in definition 7. The adaptation consists in applying the different adaptation operators of the retrieved adaptation method. The solution is obtained by adding to the solution of $\mathtt{srce}$ the variations involved by the different variables of the problem (calculated using the partial derivatives).

The knowledge acquisition process is performed according to the principle introduced before: a candidate solution produced by the system is always tested by the oracle. If the solution is not satisfactory, the involved adaptation operators are tested and corrected if needed, until a satisfactory solution is found. Then, the newly solved case $(\mathtt{tgt}, \mathtt{Sol}(\mathtt{tgt}))$ is added to the case base together with its adaptation method, given by the oracle.

The oracle is simulated by the function f and by a tolerance threshold $\varepsilon > 0$ (the maximal tolerated error). The oracle is capable of computing the distance between two solutions, to give the correct solution for a case and to give the adaptation methods.

The following example illustrates the mechanism of IakA-NF with a function $f_a : \mathbb{R} \to \mathbb{R}$. The first part of the example describes the notations used and the knowledge available in the system:

Example : Oracle knowledge, source case and target problem are defined as follows:

$$\begin{array}{rl} f_a : \mathbb{R} \to \mathbb{R} & f_a(x) = \begin{cases} 1 + \arctan(3x) \text{ if } x \geq 0 \\ -1 + \arctan(3x) \text{ if } x < 0 \end{cases} \\ \mathtt{srce} = x^s & \mathtt{tgt} = x^t \\ \mathtt{Sol}(\mathtt{srce}) = y^s & \widetilde{\mathtt{Sol}}(tgt) = \widetilde{y^t} \end{array}$$

Moreover, there is only one adaptation operator $\mathtt{AO}_r$ in the adaptation method $\mathtt{AM}(\mathtt{srce})$. It is defined by $x^s \ \mathtt{r} \ x^t$ holds for any x^s and x^t, and $\widetilde{y^t} = \mathcal{A}_{\mathtt{r}}(x^s, y^s, x^t) = y^s + \frac{\partial y^s}{\partial x}(x^t - x^s)$.

4.2 Experiments

Several experiments have been conducted with IakA-NF. Two of them are presented below.

Influence of the tolerance threshold of the oracle. The aim of this experiment is to analyze the impact of ε (the tolerance threshold of the oracle) on the

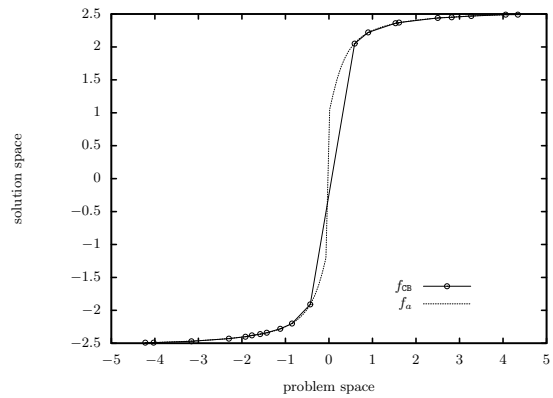

Fig. 3. Representation of the oracle knowledge f_a and of the system knowledge $f_{\mathtt{CB}}$ for an initial case base of 20 cases (circles represent cases).

quality of the results produced by the system. The hypothesis is that the smaller ε is, the better the results are (for a constant number of solved problems).

In order to conduct this experiment, an initial knowledge base is built; it consists 20 cases randomly generated (and solved by the oracle $\mathcal{O}_{f_m}$) and their associated adaptation methods (also given by the oracle $\mathcal{O}_{f_m}$).

$$\mathcal{O}_{f_m} = (f_m, \varepsilon) \qquad \begin{aligned} & f_m : \mathbb{R} \to \mathbb{R} \\ & f_m(x) = x + 10\sin(5x) + 7\cos(4x) \end{aligned}$$

Moreover, 70 target problems are randomly generated. The same initial knowledge base and set of problems are used for all the tests in this experiment.

Two systems are run in parallel: the control system and the IAKA system. The goal is to solve the 70 problems of the set of problems. In both systems, problems are solved according to the IAKA approach (test and repair of the knowledge of the system). The difference is that solved cases are not added to the case base in the control system whereas they are in the IAKA system.

The purpose of the experiment is to make ε vary, thus the experimental protocol described above is made 10 times with 10 different values for ε. For each experiment, we compare, for each case, the difference between the error made by the control system and by the IAKA system.

Two statistical tests are performed on the gathered data: the *Z-test* [8] and the *Wilcoxon test* [13] to measure the efficiency of the knowledge acquisition process. The value ρ, determined in each test, is the probability of obtaining the same results in a system performing knowledge acquisition as in a system without knowledge acquisition. For a IAKA system, the smaller ρ is, the lower the chances of obtaining such results with the control system are. Therefore, the

smaller ρ is, the better the IakA system is. The Z-test is a parametric test for two paired samples. As there is no guarantee that the initial distribution of cases and problems follows a normal law, the Wilcoxon test, a non-parametric test for two paired samples, is used to confirm the results of the first test. Figure 4 shows a graphical interpretation of the results of the Wilcoxon test (the results of the Z-test are similar).

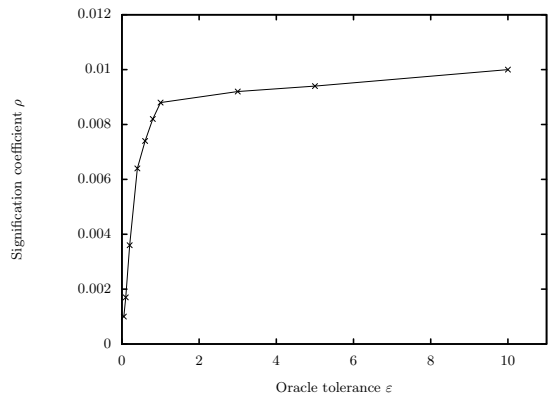

Fig. 4. Evolution of the value of ρ in function of ε for the Wilcoxon test.

On the charts, we observe that the smaller ε is, the smaller ρ is, i.e., the more effective the system is. A significant difference ($\rho < 0.01$ i.e. 1%) in terms of reduction of the size of the error is achieved when $\varepsilon = 10$ (which is a high value in this domain). The conclusion is that the higher the tolerance threshold of the oracle is, the bigger the probability for the system to make a mistake is, which confirms the hypothesis of this experiment. Similar tests have been performed with problems of two and three variables, giving similar results.

Impact of a discontinuity on the CBR process. The aim of this experiment is to analyze the behavior of a CBR system solving problems by approximation when there is a discontinuity in the domain. This experiment is motivated by the observation (b) discussed in section 3. The hypothesis is that more interactions with the oracle are needed when a problem is in the neighborhood of a discontinuity.

As for the previous experiment, an initial knowledge base of 20 cases randomly generated is built by the oracle $\mathcal{O}_{f_a}$, and 70 target problems are also randomly generated. The oracle is defined as $\mathcal{O}_{f_a} = (f_a, \varepsilon)$ with f_a as defined in section 4.1. The experiment consists in solving the 70 target problems with IakA-NF. The results are processed to count the number of problem-solving episodes that have required a correction from the oracle. As an example, figure 5

shows a graphical interpretation of the result of an experiment conducted with a tolerance threshold $\varepsilon = 0.2$.

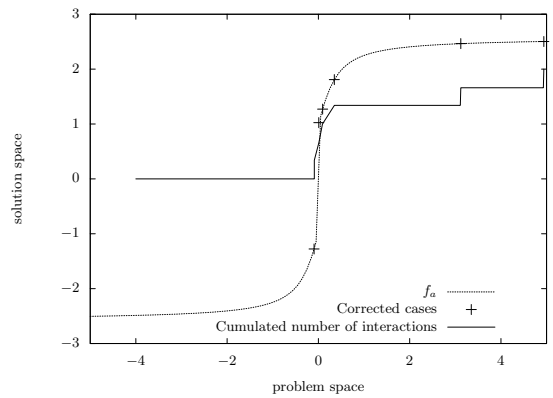

Fig. 5. Distribution of the corrected cases around a discontinuity (with $\varepsilon = 0.2$). The dotted line represents the function to approximate, the crosses are the solved cases that have required a correction from the oracle and the plain line represents the accumulation of the number of interactions with the oracle.

This experiment has been conducted several times with different values for ε (but still with the same initial knowledge base and the same series of problems). Table 1 gives the results of these experiments. Empirical results show

Value of ε	0.05	0.1	0.2	0.5	1.0	1.5	2.0	5.0	10
Number of corrected cases	20	13	6	5	3	3	2	0	0
Number of corrected cases around the discontinuity[a]	16	13	5	4	3	3	2	0	0

[a] The interval "around discontinuity points" is determined manually before the experiment

Table 1. Number of corrected cases and number of corrected cases around the discontinuity in function of the tolerance of the oracle.

that the number of cases learned around a discontinuity grows while the oracle tolerance threshold decreases. This tends to confirm the initial hypothesis of this experiment. The same experiment was also conducted with another function f_{ht} involving two problems variables.

$$\mathcal{O}_{f_{ht}} = (f_{ht}, \varepsilon) \qquad f_{ht} : \mathbb{R}^2 \to \mathbb{R}$$

$$f_{ht}(x,y) = \begin{cases} -3 - g(x,y) \text{ if } x^2 + y^2 \leq 4 \\ -g(x,y) \text{ if } x^2 + y^2 > 4 \end{cases}$$

$$g(x,y) = \sin\sqrt{x^2+y^2} + \frac{x}{7}$$

For two-dimensional problems, the results and the conclusions are similar. Figure 6 illustrates the conclusion. In this example, the oracle is $\mathcal{O}_{f_{ht}}$, $\varepsilon = 1.0$ and 20.000 problems are solved. Only 149 cases had to be corrected by the oracle, 113 of which during the first 1000 solved problems.

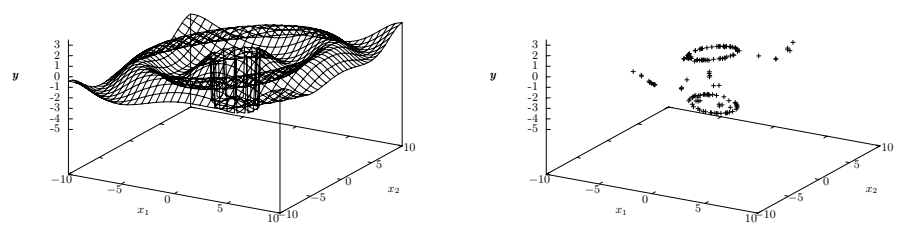

Fig. 6. Acquisition of cases around a discontinuity. The figure on the left represents the oracle knowledge. The figure on the right shows the cases learned by the system (after correction by the oracle): a high proportion of cases are acquired near discontinuity points. It must be remarked that there is a discontinuity around the top of the curve.

5 Discussion

IakA is different from off-line approaches in that the knowledge, coming from the external world, is acquired incrementally. Off-line approaches generate a large amount of knowledge at once, leading to a significant work for the domain expert to interpret the results. In IakA, the gradual acquisition alleviates the effort required by the oracle. IakA may be used as a complement of a first acquisition phase: it offers an easy way to acquire additional knowledge.

Among on-line methods, Chef learns from failures but differs from IakA in that it exploits its own knowledge to explain failures and to avoid them in further reasoning. In Dial, an adaptation case base is used to support an incomplete

adaptation rule base but it does not evolve over time. By contrast, IakA updates its existing adaptation methods whenever a failure occurs.

On-line learning in CBR is usually limited to the accumulation of cases and to their indexing. A failure due to system knowledge may reoccur several times if the involved knowledge is not corrected. In IakA, the role of the oracle is to correct such knowledge. The effort required from the oracle might seem quite important but it is limited compared to the one required in off-line methods. Moreover, this effort cannot be avoided when focusing on knowledge that usually resists other knowledge acquisition approaches.

6 Conclusion

This paper has described IakA, an approach for on-line acquisition of cases and adaptation knowledge based on interactions with an oracle (which can be considered as an "ideal expert"). IakA has been designed using the idea of a unified view of the knowledge involved in the CBR process. The failures of the CBR inference are used to repair the knowledge base (adaptation knowledge within cases). The decomposition of the adaptation process into several steps makes the identification of the knowledge involved in the failure easier. IakA-NF is a prototype for testing IakA in the domain of numerical functions with an automatic oracle. The tests show that IakA opportunistic knowledge acquisition improves the accuracy of the CBR system in the vicinity of the place where failures have occurred. They also show that this acquisition ceases to be efficient around discontinuity points, where the CBR principle is violated.

Although it has been tested, the IakA approach remains to be compared with a real-world application, using an expert instead of an oracle (where an expert can be seen as "a noisy oracle whose availability is usually quite low").

Three kinds of failure have been described in this paper. Failures of type (b) were the subject of experiments. However, additional work can be done to improve the efficiency of the knowledge acquisition. For instance, when several failures occur in the same part of the space, the system could point it out to the expert. The interaction, that takes place off-line, may lead to the explicit modelling of additional knowledge in this part of the space (e.g. "there is a discontinuity in 4"). This knowledge could then be added to the system, thus avoiding the consideration that 3.99 and 4.01 are similar in further reasoning. Failures of types (a1) and (a2) may also lead to knowledge acquisition. With regard to type (a1) failures, experiments are currently conducted to measure the impact of the addition of intermediate cases (by the oracle) when there is no similar source case. The study of failures of type (a2) is possible future work.

The IakA approach and its justification rely on the viewpoint of CBR as system producing approximate solutions. Another viewpoint is that of uncertain reasoning. A future work direction aims at generalizing the IakA approach and its justification so that it considers both viewpoints.

As discussed in section 5, IakA should inter-operate with other knowledge acquisition/extraction/learning approaches. Most of the time, these approaches

are supposed to be applicable to different phases of CBR, with different goals and with different knowledge sources. However, IAKA adopts a unified view of the CBR process and its knowledge. Therefore, more work must be done to connect the various approaches in a more general framework. For instance, a future work is to elaborate a strategy that focuses on the type of faulty knowledge (adaptation knowledge, strategic knowledge, domain knowledge, etc.) to trigger an appropriate acquisition method. Although this is a long-term future work, the authors' opinion is that this is an important issue in the field.

References

1. A. Cordier, B. Fuchs, J. Lieber, and A. Mille. Failure Analysis for Domain Knowledge Acquisition in a Knowledge-Intensive CBR System. In R. Weber M. Richter, editor, *Proceedings of the 7th International Conference on Case-Based Reasoning*, LNAI 4626. Springer, 2007.
2. A. Cordier, B. Fuchs, and A. Mille. Engineering and Learning of Adaptation Knowledge in Case-Based Reasoning. In LNAI 4248, editor, *Proceedings of the 15th International Conference on Knowledge Engineering and Knowledge Management (EKAW-2006)*, pages 303–317. Springer, Berlin, 2006.
3. S. Craw, N. Wiratunga, and R. Rowe. Learning adaptation knowledge to improve case-based reasoning. *Artificial Intelligence*, 170(16-17):1175–1192, 2006.
4. M. d'Aquin, F. Badra, S. Lafrogne, J. Lieber, A. Napoli, and L. Szathmary. Case Base Mining for Adaptation Knowledge Acquisition. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI'07)*, pages 750–755. Morgan Kaufmann, Inc., 2007.
5. D. Dubois, F. Esteva, P. Garcia, L. Godo, R. L. de Màntaras, and H. Prade. Fuzzy Modelling of Case-Based Reasoning and Decision. In D. B. Leake and E. Plaza, editor, *Case-Based Reasoning Research and Development – Second International Conference, ICCBR'97, Providence, RI, USA*, Lecture Notes in Artificial Intelligence 1266, pages 599–610. Springer Verlag, Berlin, 1997.
6. K.J. Hammond. Explaining and Repairing Plans That Fail. *Artificial Intelligence*, 45(1-2):173–228, 1990.
7. K. Hanney. *Learning Adaptation Rules from Cases.* MSc Thesis, Trinity College Dublin, Ireland, 1996.
8. M. G. Kendall and A. Stuart. *The advanced theory of statistics: Tome 1 distribution theory.* New York: Hafner, 1969.
9. D. Leake, A. Kinley, and D. Wilson. Learning to integrate multiple knowledge sources for case-based reasoning. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence.* Morgan Kaufmann, 1997.
10. David B. Leake, Andrew Kinley, and David Wilson. Learning to Improve Case Adaptation by Introspective Reasoning and CBR. In *ICCBR'95*, pages 229–240, 1995.
11. B. Smyth and M. T. Keane. Retrieving Adaptable Cases: The Role of Adaptation Knowledge in Case Retrieval. In *Lecture Notes In Computer Science; Vol. 837*, pages 209–220. Springer-Verlag - London, UK, 1993.
12. B. Smyth and M. T. Keane. Adaptation-Guided Retrieval: Questioning the Similarity Assumption in Reasoning. *Artificial Intelligence*, 102(2):249–293, 1998.
13. F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics*, 1:80–83, 1945.