



**HAL**  
open science

## A conversational case-based reasoning approach to assisting experts in solving professional problems

Negar Armaghan, Jean Renaud

### ► To cite this version:

Negar Armaghan, Jean Renaud. A conversational case-based reasoning approach to assisting experts in solving professional problems. *Knowledge Management & E-Learning: An International Journal*, 2018, 10 (1), pp.53-66. hal-02401367

**HAL Id: hal-02401367**

**<https://hal.science/hal-02401367>**

Submitted on 9 Dec 2019

**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.

---

**A conversational case-based reasoning approach to assisting experts in solving professional problems**

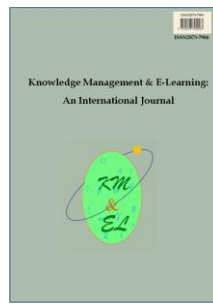
---

**Negar Armaghan**

Iranian Research Organization for Science and Technology (IROST), Tehran, Iran

**Jean Renaud**

Institut National des Sciences Appliquées (INSA), Strasbourg, France



**Knowledge Management & E-Learning: An International Journal (KM&EL)**  
ISSN 2073-7904

**Recommended citation:**

Armaghan, N., & Renaud, J. (2018). A conversational case-based reasoning approach to assisting experts in solving professional problems. *Knowledge Management & E-Learning*, 10(1), 53–66.

---

## **A conversational case-based reasoning approach to assisting experts in solving professional problems**

---

Negar Armaghan\*

Department of Technology Development Studies  
Iranian Research Organization for Science and Technology (IROST), Tehran, Iran  
E-mail: armaghan@irost.ir

Jean Renaud

Laboratoire de Génie de la Conception (LGECO)  
Institut National des Sciences Appliquées (INSA), Strasbourg, France  
E-mail: jean.renaud@insa-strasbourg.fr

\*Corresponding author

**Abstract:** Nowadays, organizations attempt to retrieve, collect, preserve and manage knowledge and experience of experts in order to reuse them later and to promote innovation. In this sense, Experience Management is one of the important organizational issues. This article is discussed the main ideas of a future Conversational Case-Based Reasoning (CCBR) intended to assist the experts of after-sales service in a French industrial company. The aim of this research is to formalize the experience of experts in after-sales service in order to better reuse them for similar problems in future. The research opts for an action research method which consists of two main parts: description of failure and proposition of decision protocol. The data were complemented by questionnaires, documentary analysis (including technical reports and other technical documents), observation and many interviews with experts. The findings include several aspects: the formalization of Problem-solving Cards, proposing the structure of case base, as well as the framework of proposed system. These formalizations permit after-sales service experts to provide effective diagnosis and problem-solving.

**Keywords:** Conversational case-based reasoning; Experience management; Problem solving; Diagnosis

**Biographical notes:** Dr. Negar Armaghan is an Assistant Professor in the Department of Technology Development Studies, at Iranian Research Organization for Science and Technology (Iran). She has been involved in multiple disciplinary researches in the area of knowledge management, experience management, case-based reasoning, change management, and problem solving and learning in organizations. She has as well published several scientific works and papers in these domains.

Prof. Jean Renaud is a Full Professor at the INSA of Strasbourg (France). He is a Director of the laboratory in the product design. His research concerns the analysis and the optimization of multi-criterion and knowledge management in manufacturing sector for the development of new products. Jean Renaud teaches particularly the project management and industrial management.

---

## 1. Introduction

Nowadays, organizations attempt to retrieve, collect, preserve and manage knowledge and experience of experts in order to reuse them later and to promote an innovative process. In this sense, Experience Management (EM) has become one of the important organizational issues. Organizations have to perform EM in order to manage their tacit and explicit knowledge (Chen & Chen, 2011; Prax, 2012). For that reason, EM could have significant influence in the learning process and problem-solving in innovative organizations (Armaghan, 2016). Effective learning in problem-solving context is difficult to achieve, because, problem-solving tasks often involve complex processes that are inaccessible to learners. It is important to make such complex processes visible for observation and practice and provide learners with necessary help during the learning process (Yuan, Wang, Kushniruk, & Peng, 2016). In this paper, we have studied this issue in an industrial company. A French industrial company, called Numalliance, decided to manage the experience of experts in after sales-service department in order to reuse the previous experience in future cases. This company is specialized in design and manufacturing of Computer Numeric Control (CNC) bending machines with numerical control for metallic wire, tubes and strip, with secondary operation. The after-sales service of this company deals with the issue of EM, and needs better problem-solving for the failure and dysfunction of machines delivered to customers by reusing previous experiences of experts. The company intends to formalize the tacit knowledge and experience of the technicians in the problem-solving process. This company produces two kinds of machines: standard machines and special machines. Standard machines are those, produced several times with already existent know-how. The special machines are those, designed and produced for first time for a specific offer. The after-sales department of this company wanted to improve the quality of its service; in order to: increase customer response time; decrease diagnosis and reparation time; reuse its experts' previous experience. For these reasons the company was prompted to better formalize its previous experience and tacit knowledge. Therefore, this company is going to achieve the following objectives:

- Providing a decision support system approach to the company. This system support will assess the need for reparation and problem-solving and also saving new creative ideas and reasoning for future problem-solving,
- Reducing the rate of errors in diagnosis.
- Making experts more independent during reparation. They will need less help from others in the problem-solving process,
- Reducing the demand rate for the same problem.
- Supporting less experienced experts or novices. Allowing novices to use this system will help them to train and learn effective approach in problem-solving. Thus, they will not necessarily need to be experts in a particular area because of the simplification of the approach by organizing questions and answers in the system,
- Making experience management and knowledge capitalization achieved systematically.
- Improving the problem-solving process to increase service quality.
- Allowing experts to have more time to solve new and complicated problems.
- Reducing the interventions of experts.

At present, these experiences are saved as unusable information in their data base. The data base is a collection of failure forms which are filled in by the experts of after-sales service after a case of problem-solving. Actually, these forms are mainly in textual format and are very difficult to be reused at a later stage. The aim of this approach is to make a correct diagnosis and to have a coherent and consistent approach, which will save time for both the expert and the customer to carry out a problem-solving in a shorter period.

Case-Based Reasoning (CBR) is an approach to solve a new problem by remembering a previous similar situation and by reusing information and knowledge from that situation (Aamodt & Plaza, 1994). It has been used to develop many systems applied in a variety of domains, including manufacturing, design, law, diagnosis and planning (Kolodner 1993; Yan, Qian, & Zhang, 2014). Principles from CBR research serve as a foundation for applied computer systems for tasks such as supporting human decision making, aiding human learning, and facilitating access to electronic information repositories (Leake, 2015).

Therefore, our objective is to find a way to put forward a system of exploitation of these failure forms and present them in a convertible format by using a Conversational Case-Based Reasoning (CCBR) System. The objective is to show the practical feasibility of this approach and its utility for industrial application. Section 2 contains background information on CCBR and also introduces the notion *conversation* by this mode of reasoning. Section 3 proposes material and method of research. Section 4 explains our results obtained from this research. Section 5 suggests a discussion point and Section 6 concludes and presents the perspective of this work.

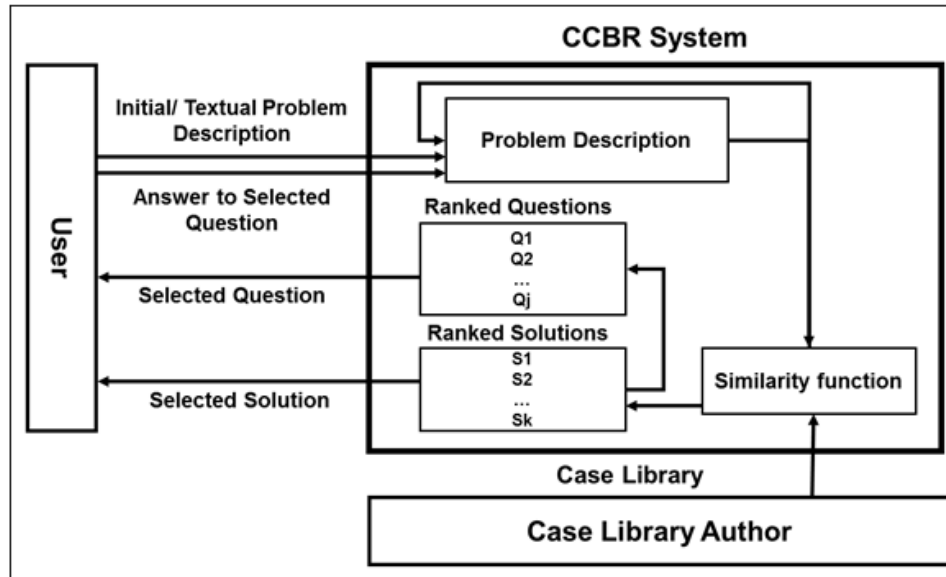
## 2. Background

CBR solves problems by using the already stored knowledge, and captures new knowledge, making it available for solving the next problem. Therefore, CBR can be seen as a method for problem-solving as well as a method to capture new experience and make it immediately available for problem-solving. It is introduced by cognitive science community and can be seen as incremental learning (Perner, 2014). A new problem is solved by seeking a similar previous case named "source case" and by reusing its solution to solve the present problem named "target case". A case is represented by the description of a problem and its solution. A "source case" is a case, which will inspire the solution of a new problem called the "target case". "Conversation" in a CBR system is a cooperative exchange between the system and its user to formulate queries for effective case retrieval and problem-solving. Consider the query formulation process for a variety of problem-solving tasks. For a diagnosis task, it involves observing and specifying symptoms and test results, while for legal reasoning it may involve describing the charges and arguments for and against the defendant. Conversation typically consists of a user identifying an initial set of features, followed by an iterative process in which the system prompts the user with a set of additional features to consider, from which the user selects one or more (Gupta & Aha, 2003, 2007; Yan, Wang, Zhang, & Zhao, 2014). The case author creates a set of cases, called a case library for the CCBR system, before problem-solving. A case  $C$  in a CCBR system is presented as follows (Aha, Breslow, & Munoz-Avila, 2001):

- **Problem**  $Case_p$  encoding the problem as:  $Case_p = Case_D + Case_{QR}$  and has the solution  $Case_s$ .
  - $Case_D$  : a short text that describes the problem  $Case_p$ ,

- $Case_{QR}$  : set of questions-answers pairs
- **Solution**  $Cases$  : a sequence of actions for responding  $Case_p$

Fig. 1 shows the general process of a Conversational Case-Based Reasoning (CCBR) system; proposed to reduce the knowledge gap between users and databases of cases in CBR systems (Aha, Breslow, & Munoz-Avila, 2001).



**Fig. 1.** The generic CCBR problem-solving process. Adapted from Aha et al. (2001)

Nowadays, the argumentation research in AI is experiencing a new reactivation. The argumentation skills increase the agents' autonomy and provide them with a more intelligent behavior (Jordan, Heras, & Julian, 2012; Heras, Jordan, Botti, & Julian, 2013b). The good results of CBR systems in argumentation domains suggest that this type of reasoning is suitable to manage argumentation processes (Goel, 1989; Heras, Jordan, Botti, & Julian, 2013a).

The CCBR systems provide ways to interact and guide users in order to refine gradually their problem description through a sequence of questions and answers. The CCBR systems interact with users during a "conversation" in order to resolve a query. A set of questions are defined in a system chosen by users and are answered during a conversation in order to find the nearest similar source case to the target problem. In problem-solving by CCBR, interaction between the system and the user is as follows (Lemontagne, 2004):

- The user provides to the system a brief textual description of a problem. The system calculates the similarity between this description and case "problem". The system offers to the user a series of questions.
- The user chooses the questions he wants to respond to. For each answer given by the user, the system will evaluate again the similarity of each case. The questions that were not answered are presented in descending order of priority.
- When the case reaches a sufficiently high level of similarity (i.e. it crosses a threshold), the system proposes a solution to this case. If any case reaches a

sufficient degree of similarity and the system has no more questions to ask the user, the problem is stored as an unresolved case.

The cases are retrieved, until, there are no more questions, or when all the retrieved cases satisfy the user. The system matches the request with memorized cases; each case includes a description of a problem (composed of a text and a series of questions-answers) and a solution. The system responds by displaying an ordered set of similar cases best matched to the target case. The list of questions-answers for each case is sorted from general cases to a particular case (Gupta, 2001). For example, for an application that helps users to solve problems related to the television remote control; it is necessary to check first "Are there any batteries in the remote control?", and then, to check "Are the batteries in the remote fresh?", Or, to verify "Are the batteries positioned correctly?" It is clear that this strict dependence requires that the last two questions should not be asked before receiving an answer to the first question. In addition, the first question should not be posed if the user has answered one of the last two questions (Gupta, Aha, & Sandhu, 2002). The solution of a case is a sequence of actions. During the conversation, the user utilizes two displays. The first display shows a ranked list of questions. The user selects questions related to the problem and answers them. Each answer to a question updates the query and ranking in both displays. The second display shows a list of cases classified by similarity to the user's query. These cases change according to the similarity of retrieved target cases to the source case. The user can select one case from this list for retrieve. The conversation will end when the user chooses a case (Aha, Maney, & Breslow, 1998).

This paper tries to structure the problem-solving process for formalizing of tacit and explicit knowledge by using CCB. The next section describes the method used for this study including the description of failures in the company.

### **3. Material and methods**

#### *3.1. Description of failures*

In this step, we will present the methodology and approaches proposed for the development of a system for aid in the diagnosis and problem-solving process. The description of failures has been done in four steps as follows.

Identifying problems and sub-problems and their classification. This phase is based on realization of a state of art in the company. It is the identification of different types of problems and it carries out a statistical study of all the failures and dysfunctions in the company. The statistical study allows us to classify the most commonly encountered failures. Our hypothesis is to define recurrent problems as a priority.

A second study identifies all components of a specific machine, called sub-problems. A statistical study provides a ranking of the most defective components. The information about this state of art is described as below:

- A phenomenological approach: consists of all the observed phenomena such as: 1- symptoms observed by the customer and provided to the expert, and, 2- interviews with experts (extracting the tacit knowledge)
- A documental approach: consists of a set of information related to technical documents, such as: 1- cards or documents, statistical studies about failure

which are related to explicit knowledge and, 2- digital display and technical documents related to parts manufacturers.

Classification of problems and components. Following the first study explained in the previous paragraph, we group said problems and components by category, called "theme". This group is complex because a problem may include more than one defective component, and a component may demonstrate several problems. This classification in themes should simplify said redundancies. Establishing links among components and problems. Once the components are known, common problems among these components are identified. These problems are considered as major problems or most commonly encountered themes. Then, for each problem, a complete description is provided.

Identifying the Symptoms. We define several symptoms for diagnosis of each problem. These symptoms are classified in two types: (1) symptoms described by the customer (the user of the machine) and, (2) symptoms described in technical documentation.

Symptoms described by the customer are all symptoms encountered by a customer when a failure or dysfunction occurs. Customers may reveal symptoms by phone, fax or email. Customers give their observations about the problem as well as their opinion or hypothesis. When a failure occurs on a customer's machine, there could be one or more symptoms reported by the customer. Generally, the customer calls the company in order to solve their problem. Normally, as the customer is not a field expert, they cannot report simultaneously all symptoms of dysfunction or failure of the machine. The customer may reveal symptoms gradually, aided by an expert. Indeed, when a customer reports the first encountered symptom, an expert will try to ask them some questions in order to acquire more information about the failure or dysfunction, for better clarifying the nature of the problem. Symptoms present on display (information from sensors) are all symptoms defined in a machine's software when a failure appears. These symptoms may appear on the screen or by an indicator on the machine. Symptoms described in the technical documentation are symptoms provided by machine components suppliers.

In this step we define a combination of symptoms that may cause a problem. If a general symptom calls as (S), and there are four symptoms (S1, S2, S3, S4) for a problem "A", then the combinations of symptoms will generate the problem "A": For example:  $S1 \wedge S2 \Rightarrow$  Problem A, or,  $S1 \wedge S3 \wedge S4 \Rightarrow$  Problem A.

All combinations of these two types of symptoms will allow the expert to identify the problem. That means, at least one of the combinations of above symptoms will guide a technician to diagnose the problem "A". Thus, the researchers have identified all recommendations made by experts in a failure situation. These are the recommended solutions to the current failure or dysfunction, also, the origin of failure. Then, it is necessary to describe the relation between symptoms in order to find the problem. In this stage, a chart is prepared to create and show these relations. This chart records the following data: problem, origin of failure (cause), symptoms, suggested solutions.

### *3.2. Formalization of tacit knowledge and proposition of diagnostic by decision protocols*

The final objective is formalization of the diagnostic and problem-solving procedure. A diagnosis is built through a series of questions and answers (conversations) between an expert and a customer. Some questions are also followed by a check and test. In practice, during a conversation, an expert uses their experience as well as all documents they have



at their disposal in order to reduce the hypothesis made for the current problem. The customer describes their problem using everyday language to transfer information concerning the problem. Customer's explanations are mainly based on their observations when a problem occurs. The customer often provides dispersed, non-formal and sometimes incorrect data or information; and often is not familiar enough with the machine. An expert should know how to ask proper details about problem. The expert begins to ask questions when they need clarification about information provided by the customer. A good expert is someone who asks a few questions and is able to deduce the remaining information from explanations provided previously by the customer (Armaghan, 2014). The expert must guide the customer by asking the right questions at the right time. For example: the expert will ask the customer to perform a test. The customer will run a component, replace parts, etc. until the response is "it now works" or "the machine is still broken down". A less experienced expert or novice may not understand the customer well enough in order to guide them correctly and precisely. Thus, the proposed method allows experts to ask relevant questions, step by step, during the conversation. This method helps the user of the system, namely the expert, to supervise exchanges and conversations in an interactive and conversational manner. Each result obtained by conversation has the following characteristics:

- It may be a starting point to clarify another problem. The expert must create new hypotheses with new symptoms.
- It eliminates or reduces the number of hypothesis,
- Eventually, the diagnosis/problem-solving process is either successful or the problem is not solved.

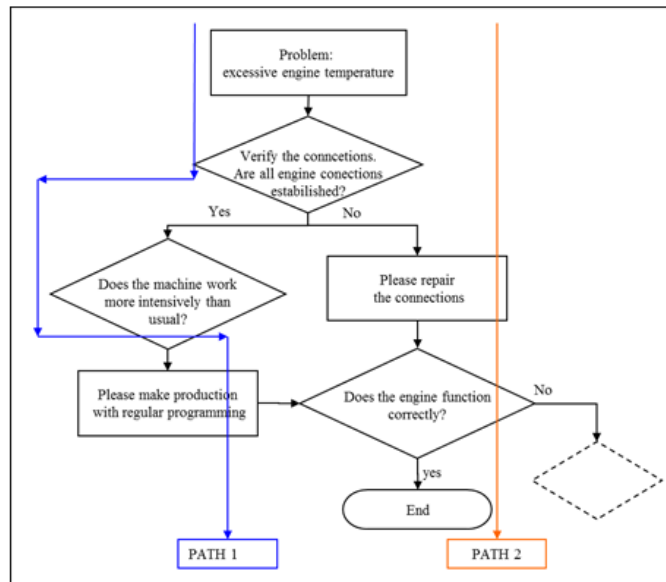


Fig. 2. Decision protocol and identification of a path in problem-solving process

The objective of the method is structuring these exchanges between experts and customers. The principle of fault tree is applied to diagnose a failure or dysfunction (Limnios, 2005, 2007; Yan, Wang, Zhang, & Zhao, 2014). A fault tree is built by a series of questions with "yes" or "no" answers; this combination is named the *conversations*. According to the answer of the question, the next branch of the fault tree is reached by

asking another question, giving another instruction, or through diagnosis. This principle is used between the customer and the expert in problem-solving process (Fig. 2). All solutions proposed for solving a general problem named a "decision protocol" are presented in Fig. 2.

Due to the minimization of the number of symbols to use in a fault tree, this way is easier to use in order to perform the diagnosis and reparation more quickly. Elimination of certain symbols such as "or" and "and" is possible thanks to the criteria that we have integrated into the fault tree during its construction. These criteria are: (1) Ease of tests (action A is easier to test than action B), (2) Plausibility (based on the experience of the expert, for example: according to the expert part A is more defective than part B). The proposed technique doesn't just make the diagnosis, but also suggests solutions for the current problem.

In a problem-solving process, it is possible to have different reasoning "paths" to follow until finding the appropriate solution (Armaghan & Renaud, 2012). Each decision protocol could have one or several paths. A path in the decision protocol consists of reasoning arguments (Fig. 2). We may have different problem-solving paths for a problem (Watson, 1999).

## 4. Results

### 4.1. Representation and formalization of a problem-solving card (PSC)

We will determine a reference case base. This case base contains a list of failures, diagnoses ways and paths of problem-solving. It is also possible that experts' experiences are available in the case base.

If "pb" is a problem (resp. "sol" is a solution) then pb (resp., sol) is a format of knowledge representation that presents a problem (resp., a solution) of this domain. There is a case base (called source case), which means a set of couples  $srce\text{-}case = (srce, Sol(srce))$  which "srce" is a problem, Sol(srce) is a solution, and, srce is for a solution Sol(srce) (Baader, Calvanese, McGuinness, Nardi, & Patel-Schneider, 2003; Becattini, Borgianni, Cascini, & Rotini, 2012).

The CD is a knowledge base which contains the knowledge of a domain. A problem  $pb$  encodes a set of special situations called instances of the problem  $pb$ . If  $pb$  and  $pb'$  are two problems, we can say that the first is more specific than the second (denoted by  $pb \models_{CD} pb'$ ), it encodes a set of instances included in the set of instances encoded by the second. Reasoning from case base is solving a problem, called the *target problem* and denoted by "target", used by the base case. This reasoning usually consists of two main steps: the *retrieval*, which means to select a *source case*, considered similar to the target problem, and *adaptation* that aims to solve a target problem by relying on the *retrieved source case*.

The current problem is gradually specified during a conversation. The conveyed information from expert to customer (resp., from customer to expert) is noted  $expert_{t=i}$  (resp.,  $Customer_{t=i}$ ). After,  $customer_{t=i}$  is both, as a textual and formal representation of a message which could be manipulated by a reasoning. We can consider that the initial problem,  $srce_0$  is a very general problem "I have a failure or dysfunction". Then each answer  $customer_{t=i}$  ( $i > 0$ ) allows the expert to specify:  $srce_i = srce_{i-1} \wedge customer_{t=i}$ . Note that in this model, the information given by the customer is not questioned, unless it falls

into contradiction: if  $srce_i$  doesn't satisfy, the construction of the problem will be revised (and some answers suggested by the customer will be revised).

The last customer response must indicate that the problem is solved (exp.  $Customer_{t=i}$ ) and not lead to a new problem (there is no defined  $srce_{i+1}$ ). A solution  $Sol(srce_i)$  of pb ( $srce_i$ ) is a message that the expert gives to the customer:  $Sol(srce_i) = expert_{t=i}$ . There is generally a question, or, an instruction accompanied by a question. Thus, a Problem-solving Card (PSC) contains a set of cases ( $srce_i, Sol(srce_i)$ ); knowing that, problem  $srce_i$  specifies the problem  $srce_{i-1}$ :  $srce_i \vdash_{CD} srce_{i-1}$ . Therefore, if  $f$  is the final number of problems (for example:  $f = 3$ ), then  $srce_f \vdash_{CD} srce_i$  for all  $i \leq f$ . The card will be indexed by its final problem  $srce_f$ .

We explained in a previous section that in problem-solving processes, we can have different paths to follow until solving the problem of customers. A path in decision protocols is a resolution path. We can have several paths for a specific problem. For example, we have previously explained two different paths to solve a specific problem.

A Problem-solving Card (PSC) is built by experts progressively by asking questions from customers and using the system. A PSC is composed of sets of cases. Indeed, in our method, we assume that each question-answer is a case. Thus, another basic assumption of our methodology is: a case in CCBP is a path. That means, a way for reasoning diagnostics. We formalize each case as PBC by keeping records of a case. During construction of PSC, the target problem is more and more accurate until obtaining a solution. In a PSC, experts can save their own explanations or interpretations, if necessary, after each question-answer. The PSC will be personalized for a specific problem for further utilization. These PSCs are able to be retrieved and reused later in similar situations by experts. For each failure, experts can study PSCs similar to the target problem. Experts can choose the most adapted card to the target case, and then present diagnosis.

#### 4.2. The framework of the proposed system

Expert knows the target $_i$ , in instance  $i$  of interaction with a customer. If a PSC corresponds to a more specific situation than the target $_i$ , in other words, if it is indexed by  $srce_f$  such as  $srce_f \vdash_{CD} target_i$ , it is possible to match it exactly or approximately to the customer's problem, once it has been completely specified. The framework of the system is to provide to the expert, at any time  $i$ , all of these PSCs. The expert then decides to reuse it (or not). The framework of proposed system is shown as following algorithm:

*Start*

Target $_0 \leftarrow$  «I have a failure or dysfunction. »

Sol(target $_0$ )  $\leftarrow$  « Hello, could you please specify it? »

$i \leftarrow 0$

Repeat

    Ask expert to read the message Sol (target $_i$ ) to customer (or a similar message).

$i \leftarrow i + 1$

    either customer $_{t=i}$  customer's response.

    if customer $_{t=i} \neq$  OK

        then target $_i \leftarrow$  target $_{i-1} \wedge$  customer $_{t=i}$

        Or  $I$  set of "Indexed" PSC by  $srce_f$  such as  $srce_f \vdash_{CD} target_i$ .

        Give to user an access to  $I$ , for being able to use it, if necessary, in order to assist user to formulate the next message for customer: Sol (target $_i$ ).

*Till OK*

Save the new PSC indexing by the target problem<sub>i-1</sub>  
*End.*

#### *4.3. Structure of case base*

The structure of case base was developed by leveling information and classifying problems. Each problem was identified by some symptoms. Problem and its class were identified by using the symptoms. Then, the procedures of problem-solving were presented from decision protocols. On the top, there is the problem or symptom and progressively, solution of problem is specified in descending. Indeed, we can say a path was created from problem to solution. This path was made by all the questions and answers.

The structure of case base in proposed system consists of all PSCs. This system is interactive which stands for two objectives: 1- to capture and save the cards, and, 2- to suggest similar cards to the current problem. At the beginning, we have a case base with a scratch card. The case base starts to gradually expand, when an expert solves a problem and creates a new card and saves it in the system. The system starts with a very general question. At each stage, the expert asks a question or gives an instruction in order to better define the problem. The first question of the expert is based on information (symptoms) that the customer provides to him. When the expert obtains the necessary information from the customer, he starts proceeding and asks the questions by using the system.

The purpose of diagnosis is to minimize the number of tests needed to reach a proper diagnosis and thus reduce the risks and costs associated to testing. Another important point to be considered by the system designers is the acceptability of the system by users. There are various forms of a diagnostic process (Mesherry, 2001): mixed-initiative dialogue, coherent and logical dialogue, relevant dialogue, explanation of reasoning, tolerance for incomplete data, sensitivity analysis. Some CCBR systems rely on a "mixed-initiative dialogue" where the user can select the questions from a list of questions and not only select the most used question in the system (Aha, 1998; Aha, Maney, & Breslow, 1998; Watson, 1999). It is important to allow experts (users of system) to note personal opinion or solutions, not only because they are more acceptable to experts, but also, to increase and improve efficiency in future problem-solving processes. It is possible that the expert retrieves a previous similar case related to personal experience and recognizes what descriptors of problems are more important and relevant than others. Another aspect of "mixed-initiative dialogue" is whether the expert can propose personal "opinion" or suggestion. A system that does not take into account the expert's opinion misses an opportunity to include their experience in the problem-solving procedure. This may generate inefficient problem-solving. Furthermore, in diagnostic processes it is possible for a professional expert (the expert in a specific field) to have a good idea of the problem. Thus, the expert can consider the system as a way to confirm their opinion. Updating systems continuously by experts is a way to provide an incremental experience management in CCBR.

Therefore, we rely on the idea of "mixed-initiative dialogue" in our proposed system. It is important to save the "opinion" of experts in problem-solving, in order to reuse their point of view and ideas in further similar problem-solving cases. Thus, we suggest that experts can select the questions from a list in the proposed system. The selection of questions continues until the expert identifies the problem clearly. When the problem is well defined, the system will offer a specific place called "comments" to record any additional information for each question-answer. This information may

include: the expert's opinion, explanations for a special case, and/ or any information that can help future experts in a similar problem-solving case.

#### 4.4. Adaptation guided retrieval

The objective of the retrieving step is to find and propose a solution closest to the target problem. A similar solution will require a little adjustment. It is a solution that will be the easiest to adapt. In order to find a solution to a target problem, the descriptor of the problem must be linked explicitly to the descriptor of the solution. This linking can be carried out by using the adaptation knowledge<sup>1</sup>. The descriptors play somewhat important roles in problem-solving; in particular, some descriptors of the problems have a greater influence on the solution. The retrieve approach that uses the adaptation knowledge for remembering similar cases is called "Adaptation Guided Retrieval". It consists of using adaptive knowledge when retrieving in order to determine which descriptors of the problems have greater influence on the solution (Chebel-Morello, 2008; Fuchs, 2008). The objective of adaptation is to minimize "the adaptation effort". Therefore, adaptation operations should be minimized by choosing the cases that will be the easiest to adapt (Smyth & Keane, 1996; Fuchs, Lieber, Mille, & Napoli, 2006; Matta, 2008; Fuchs, 2008).

The other hypothesis in this research was: the descriptors of case in the retrieving process are the symptoms which are defined in order to identify the problem. The described system does not include an adaptation process: this is done by experts. This system is established according to the framework presented above and will execute in a "semi-automatic" way, only the retrieving step. That means the system selects and offers several PSCs. Hence, the final selection is made by the expert. If the principle of Adaptation Guided Retrieval is followed, it will only perform the retrieving step. In other words, in the retrieving step, one source case is preferred to another, if it needs less "adaptation effort" to solve the target problem.

## 5. Discussion

The results show that we can formalize the experience and knowledge of experts by using the conversational case-based reasoning and the fault tree. We have determined a set of symptoms to diagnose and identify the problem. We have identified appropriate descriptors for elaboration of a case in the representation and formalization phases. The descriptors of problems are important for searching in the case base. We have first created the decision protocols in order to model our conversational cases. In this study, fifteen decision protocols were carried out and obtained. Each of these contains several paths in the problem-solving process.

A computer model for the future system of diagnosis is proposed. This model is constructed from the problem-solving cards in the case base. Therefore, we have established an Adaptation Guided Retrieval to use similar cards in a new situation of failure. An algorithm is proposed to store new cards by indexing the solution of problems resulting from the last conversational case (repair or replacement of components). The role of indexing is to insure the most efficient retrieve based on the descriptors of

---

<sup>1</sup> In general, in adaptation, prior similar solution must be modified to take into account the differences in order to match the new situation. The modified knowledge to carry out the reuse step is considered the "*reuse knowledge*."

problems which are more important in the problem-solving process. Indexing is used in the retrieving step in order to quickly locate a category for a target problem to be filed in. We classify the problem-solving cards in the computer model proposed by defective machine components. The system users can also save individual opinions after each conversation during the creation of cards.

We have created a very interactive system: the retrieving step is semi-automatic, and the adaptation is entirely made by experts. The automatic part of the retrieval allows experts to clarify the category of cards through interaction with customers until finding the main problem. Adaptation is "manual", which means that the expert can have these cards continue the interaction with the customer. The closest card is the one that needs the least "adaptation effort". In other words, the card that helps the most experts in a minimum time to propose a solution for the current problem.

We have also considered the following aspects in the proposed computer model:

- We avoid asking for textual description of a problem during presentation of the problem. That means that we start asking questions or giving instructions from the first step of diagnosis in order to clarify the problem.
- Our proposition allows experts to consider individual opinion in the problem-solving process. On one hand, the opportunity to choose questions is given to the expert, and on the other hand, once the problem is specified, a backup of their individual opinion is possible in problem-solving cards.

## 6. Conclusion

Tacit and explicit knowledge extractions through several individual and group interviews permitted us to analyze the problem of experience management in a real company. We could modify the classification of problems in company and propose a new classification based on the recurrent problems by linking them to the components. We also could formalize tacit knowledge of experts in order to offer them a new problem-solving approach through decision protocols and their formalization through CCBR.

This study also has some limits. This study needs to be adapted to the new generation and technology of machines. However, we should specify that the knowledge extraction and formalization of all experts with different expertise and knowledge domain is not always a given. Also, data collection from company agents is not very easy to be done. Indeed, when employees are not aware about experience management, there is a high risk of losing information and knowledge from different points of view of several experts. The construction of database is not easy compared to all the logical problem-solving methods. From a practical point of view, it is also necessary to maintain regularly the data base and update information.

In perspective, we could imagine a hybrid system that integrates a rule-based system in a CCBR system. Our proposed model was able to cope with a real situation in a company; however, it is possible to develop the selection of descriptors further in order to completely achieve Adaptation Guided Retrieval.

## References

- Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1), 39–59.

- Aha, D. W. (1998). The omnipresence of case-based reasoning in science and application. *Knowledge-Based System, 11*(5/6), 261–273.
- Aha, D. W., Breslow, L. A., & Munoz-Avila, H. (2001). Conversational case-based reasoning. *Applied Intelligence, 14*(1), 9–32.
- Aha, D. W., Maney, T., & Breslow, L. A. (1998). Supporting dialogue inferencing in conversational case-based reasoning. *Lecture Notes in Computer Science, 1488*, 262–270.
- Armaghan, N. (2014). Un cas de retour d'expérience dans les projets industriels machine-outil du point de vue académique. In J. Stal-LeCardinal, J. L. Giordano, & G. Turré (Eds.), *Les retour d'expérience du projet, réduire les risques, augmenter les performances collectives* (pp. 199–219). Paris, France: AFNOR.
- Armaghan, N. (2016). Analysis of failure challenges in implementation of knowledge management projects in Iran. *Quarterly Journal of Industrial Technology Development, 14*(27), 27–38.
- Armaghan, N., & Renaud, J. (2012). An application of multi-criteria decision aids models for case-based reasoning. *Information Sciences, 210*, 55–66.
- Baader, F., Calvanese, D., McGuinness, D., Nardi, D., & Patel-Schneider, P. (2003). *The description logic handbook: Theory, implementation and applications*. Cambridge, UK: Cambridge University Press.
- Becattini, N., Borgianni, Y., Cascini, G., & Rotini, F. (2012). Model and algorithm for computer-aided inventive problem analysis. *Computer-Aided Design, 44*(10), 961–986.
- Chebel-Morello, B. (2008). Enjeux et processus du retour d'expérience. In J. Renaud, E. Bonjour, B. Chabell-Morello, & N. Matta (Eds.), *Retour et capitalisation d'expérience: Outils et démarche*. La Plaine Saint-Denis: AFNOR.
- Chen, M. Y., & Chen, C. C. (2011). Options analysis and knowledge management: Implications for theory and practice. *Information Sciences, 181*(18), 3861–3877.
- Fuchs, B. (2008). Raisonement à Partir de Cas. In J. Renaud, E. Bonjour, B. Chabell-Morello, & N. Matta (Eds.), *Retour et capitalisation d'expérience: Outils et démarche*. La Plaine Saint-Denis: AFNOR.
- Fuchs, B., Lieber, J., Mille, A., & Napoli, A. (2006). Une première formalisation de la phase d'élaboration du raisonnement à partir de cas. In *Proceedings of the 14e Atelier de Raisonement à Partir de Cas*. Besançon, France.
- Goel, A. K. (1989). *Integration of case-based reasoning and model-based reasoning for adaptive design problem-solving*. Doctoral dissertation, Ohio State University, USA.
- Gupta, K. M. (2001). Taxonomic conversational case-based reasoning. *Lecture Notes in Computer Science, 2080*, 219–233.
- Gupta, K. M., & Aha, D. W. (2003). A framework for incremental query formulation in mixed-initiative case-based reasoning. In *Proceedings of the ICCBR-03 Workshops* (pp. 172–180). Trondheim, Norway.
- Gupta, K. M., & Aha, D. W. (2007). Conversation for textual case-based reasoning. In *Proceedings of the Workshop on Textual Case-Based Reasoning: Beyond Retrieval, ICCBR-07*. Belfast, Northern Ireland.
- Gupta, K. M., Aha, D. W., & Sandhu, N. (2002). Exploiting taxonomic and causal relations in conversational case retrieval. *Lecture Notes in Computer Science, 2416*, 133–147.
- Heras, S., Jordan, J., Botti, V., & Julian, V. (2013a). Argue to agree: A case-based argumentation approach. *International Journal of Approximate Reasoning, 54*(1), 82–108.
- Heras, S., Jordan, J., Botti, V., & Julian, V. (2013b). Case-based strategies for argumentation dialogues in agent societies. *Information Science, 223*, 1–30.

- Jordan, J., Heras, S., & Julian, V. (2012). Argumentation tool that enables agents to argue. In *Proceedings of the First International Conference on Agreement Technologies*. Dubrovnik, Croatia.
- Kolodner, J. L. (1993). Maintaining organization in a dynamic long-term memory. *Cognitive Science*, 7(4), 243–280.
- Leake, D. (2015). Problem solving and reasoning: Case-based. In J. D. Wright (Ed.), *International Encyclopedia of the Social & Behavioral Sciences* (2nd ed., vol. 19; pp. 56–60).
- Lemontagne, L. (2004). *Une approche CBR textuel de réponse au courrier électronique*. Thèse de doctorat, Université de Montréal, Montréal, Canada.
- Limnios, N. (2005). *Arbres de défaillances* (2nd ed.). Paris, France: Hermes Science-Lavoisier.
- Limnios, N. (2007). *Fault trees—Control systems, robotics and manufacturing series*. Hoboken, NJ: John Wiley & Sons.
- Matta, N. (2008). Capitalisation des connaissances et mémoire de projets. In J. Renaud, E. Bonjour, B. Chabell-Morello, & N. Matta (Eds.), *Retour et capitalisation d'expérience: Outils et démarche*. La Plaine Saint-Denis: AFNOR.
- Mcsberry, D. (2001). Interactive case-based reasoning in sequential diagnosis. *Applied Intelligence*, 14(1), 65–76.
- Perner, P. (2014). Mining sparse and big data by case-based reasoning. *Procedia Computer Science*, 35, 19–33.
- Prax, J. Y. (2012). *Le manuel du knowledge management* (3e éd.). Paris: Polia Editions Dunod.
- Smyth, B., & Keane, M. T. (1996). Using adaptation knowledge to retrieve and adapt design cases. *Knowledge-Based Systems* 9(2), 127–135.
- Watson, I. (1999). Case-based reasoning is a methodology not a technology. *Knowledge - Based Systems*, 12(5/6), 303–308.
- Yan, A., Qian, L., & Zhang, C. (2014). Memory and forgetting: An improved dynamic maintenance method for case-based reasoning. *Information Sciences*, 287, 50–60.
- Yan, A., Wang, W., Zhang, C., & Zhao, H. (2014). A fault prediction method that uses improved case-based reasoning to continuously predict the status of a shaft furnace. *Information Sciences*, 259, 269–281.
- Yuan, B., Wang, M., Kushniruk, A. W., & Peng, J. (2016). Design of a computer-based learning environment to support diagnostic problem solving towards expertise development. *Knowledge Management & E-Learning*, 8(4), 540–549.