Volume 3, Issue 12, December 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Incremental Dynamic Case Based Reasoning and Multi-Agent Systems (IDCBR-MAS) for Intelligent Tutoring System

Abdelhamid Zouhair
LIST, FST of Tangier, Morocco

El Mokhtar En-Naimi
LIST, FST of Tangier, Morocco

Benaissa Amami
LIST, FST of Tangier, Morocco

Hadhoum Boukachour
LITIS, Le Havre University
Le Havre, France

Patrick Person
LITIS, Le Havre University
Le Havre, France

Cyrille Bertelle
LITIS, Le Havre University
Le Havre, France

Abstract— In this paper we present our approach in the field of Intelligent Tutoring System (ITS), in fact there is still the problem of knowing how to ensure an individualized and continuous learners follow-up during learning process, indeed among the numerous methods proposed, very few systems concentrate on a real time learners follow-up. Our work in this field develops the design and implementation of a Multi-Agent Systems Based on Dynamic Case Based Reasoning which can initiate learning and provide an individualized follow-up of learner. This approach involves 1) the use of Dynamic Case Based Reasoning to retrieve the past experiences that are similar to the learner's traces (traces in progress), and 2) the use of Multi-Agents System. Our Work focuses on the use of the learner traces. When interacting with the platform, every learner leaves his/her traces in the machine. The traces are stored in database, this operation enriches collective past experiences. The traces left by the learner during the learning session evolve dynamically over time; the case-based reasoning must take into account this evolution in an incremental way. In other words, we do not consider each evolution of the traces as a new target, so the use of classical cycle Case Based reasoning in this case is insufficient and inadequate. In order to solve this problem, we propose a dynamic retrieving method based on a complementary similarity measure, named Inverse Longest Common Sub-Sequence (ILCSS). Through monitoring, comparing and analyzing these traces, the system keeps a constant intelligent watch on the platform, and therefore it detects the difficulties hindering progress, and it avoids possible dropping out. The system can support any learning subject. To help and guide the learner, the system is equipped with combined virtual and human tutors.

Keywords— Intelligent Tutoring Systems (ITS), Multi-Agent Systems (MAS), Incremental Dynamic Case Based Reasoning (IDCBR), Similarity Measure, Traces.

I. INTRODUCTION

E-learning is a computer system which offers learners another intermediate of learning. Indeed it allows learner to break free from the constraints of time and place of training. They are due to the learners availability. In addition, the instructor is not physically present and training usually happens asynchronously. However, most E-learning platforms allow the transfer of knowledge in digital format, without integrating the latest teaching approach in the field of education (e. g. con- structivism [23], ...). Consequently, in most cases distance learning systems degenerate into tools for downloading courses in different formats (pdf, word ...). These platforms also cause significant overload and cognitive disorientation for learners. Today, it is therefore necessary to design and implement a computer system (i. e. Intelligent tutor) able to initiate the learning and provide an individualized monitoring of the learner, who thus becomes the pilot of training. The system will also respond to the learner's specific needs.

Solving these problems involves first, to understand the behavior of the learner, or group of learners, who use platform to identify the causes of problems or difficulties which a learner can encounter. This can be accomplished while leaning on the traces of interactions of the learner with the platform, which include history, chronology of interactions and productions left by the learner during his/her learning process. This will allow us the reconstruction of perception elements of the activity performed by the learner.

We consider a system Intelligent Tutoring Systems (ITS), that is able to represent, follow and analyze the evolution of a learning situation through the exploitation and the treatment of the traces left by the learner during his/her learning on the platform. This system is based, firstly on the traces to feed the system and secondly on the reconciliation between the course of the learner (traces in progress) and past courses (or past traces). The past traces are stored in a database. Our system is able to represent, follow and analyze the evolution of a learning situation through the exploitation and the treatment of the traces left by the learner during his/her learning on the platform. The analysis of the course must be executed continuously and in real time which leads us to choose a Multi-Agent architecture allowing the implementation of a dynamic case-based reasoning. Recently, several research works have been focused on the dynamic case based reasoning in order to push the limits of case based reasoning system static, reactive and responsive to users. All these works are based on the observation that the current tools are limited in capabilities, and are not able of evolving to fit the non-anticipated or emerging needs. Indeed the reuse of past experiences causes several problems, such as:

- Modeling: formalization of experience acquired (cases), indeed a few CBR systems are able to change over time the way of representing a case [6]. According Alain Mille, a case has to describe its context of use, which is very difficult to decide before any reuse and can change in time [22].
- Reuse: the selection of past experiences which are similar to the current situation.
- Treatment: the use of the classic reasoning cycle is insufficient and inadequate in dynamic or emerging situations, unknown in advance.

In order to deal with this issue, we propose a Dynamic Case Based Reasoning based on a dynamic retrieve method, and we propose a dynamic retrieving method based on a complementary similarity measure, named Inverse Longest Common Sub-Sequence (ILCSS).

The rest of this paper is organized as follows: In the second section, we present a general introduction of intelligent tutoring systems. The third section is devoted to the presentation of the analysis and decomposition needs of our architecture. In the following section, we will describe the approach of Case-Based Reasoning and in the next part, we present a Multi-agent Case-Based Reasoning, in the following part, we will propose the description of our approach in Case Based Reasoning and intelligent tutoring systems field: Incremental Dynamic Case-Based Reasoning founded on Multi-Agent System. In the nest section, we present modeling of our system. Finally, we will give the conclusion and our future work.

II. Intelligent Tutoring Systems

Intelligent Tutoring Systems (ITS) are computer systems designed to assist and facilitate the task of learning for the learner. Who can personalize learning for learners, providing a less expensive solution for a diverse generation of learners. They have expertise in so far as they know the domain knowledge, how to teach (pedagogical knowledge) and also how to acquire information about the learner. We note that, the general architecture of Intelligent Tutoring Systems was represented in our articles [10]. Several researches concerned with the design and implementation of Intelligent Tutoring Systems to assist a learner in learning. There are, for example, tutors or teaching agents who accompany learners by proposing remedial activities [11]. There are also the agents of support to the group collaboration in the learning [7] encouraging, the learners participation and facilitating discussion between them. Other solutions are based on Multi-Agent System that incorporate and seek to make cooperation among various Intelligent Tutoring Systems [5]. The Baghera platform [31] exploits the concepts and methods of Multi-Agent approach. Baghera assists learners in their work solving exercise in geometry. They can interact with other learners or teachers. The teachers can know the progress of the learners work in order to intervene if needed. These tools of distance learning do not allow an individualized, continuous and real-time learners follow-up. They adopt a traditional pedagogical approach (be-haviorist) instead of integrating the latest teaching approaches (constructivism and social constructivism [23], [30]). Finally, given the large number of learners who leave their training, the adaptation of learning according to the learners profile has become indispensable today.

Our contribution in these important areas is to design and develop an adaptive system able to ensure an automatic and a continuous monitoring of the learner. Moreover, our system is open, scalable and generic to support any learning subject.

III. Our architecture: Analysis and decomposition needs

One of the main objectives of the individualized monitoring of the learner is to envisage, to anticipate and to reduce the number of dropping out, which makes us seek a flexible and adaptive solution [9]. The complexity of the situations to be treated leads us to choose an approach based on a MAS, able to cooperate and coordinate their actions to provide a pedagogical adaptation for the learners profile.

We reconcile the problems of the analysis of the traces left by the learners activity in platform, and the decision support systems, able to represent, follow in real-time and analyze the evolution of a dynamic situation. Such a system must represent the current situation, take into account the dynamic change of the current situation, predict the possible evolution of this situation, and react depending on the particular situations and the learners profiles. This can be done by using past situations which consequences are known. It is then a question of reasoning by analogy. This type of reasoning can allow solving new problems, using already solved problems available in memory. The system we propose, allows to analyze the learners course (trace) in order to anticipate a possible dropping-out. The learning activities past traces will be the source of knowledge for the learning adaptation process, they are stored in a database called base of scenarios. Each scenario contains all determining aspects in its development, i.e, the facts that have played an effective role in the way the events proceeded. The analysis of the current situation must be continuous and dynamic. Indeed, the target case is a plot that evolves, therefore the system must take this incremental evolution into account.

IV. Case-Based Reasoning

Case-Based Reasoning (CBR) is an artificial intelligence methodology which aims at solving new problems based on the solutions of similar past problems (past experiences) [13]. The solved problems are called source cases and are stored in a case-base (base of scenarios). The problem to be solved is called target case. A CBR is a combination of knowledge and processes to manage and re-use previous experience.

The Case-Based Reasoning cycle is composed of five steps as given in Figure Fig.1:

- Presentation: the current problem is identified and completed in such a way that it becomes compatible with the contents and retrieval methods of the case-base reasoning.

- Retrieve: The task of retrieve step is to find the most similar case or cases to the current problem in the case-base.
- Reuse: The goal of the reuse phase is to modify the solution of source case found in order to build a solution for the target case.
- Revise: The phase of revision is the step in which the solution suggested in the previous phase will be evaluated. If the solution is unsatisfactory, then it will be corrected.
- Retain: retaining the new experience and add it to the knowledge-base (case-base) [12], [1].

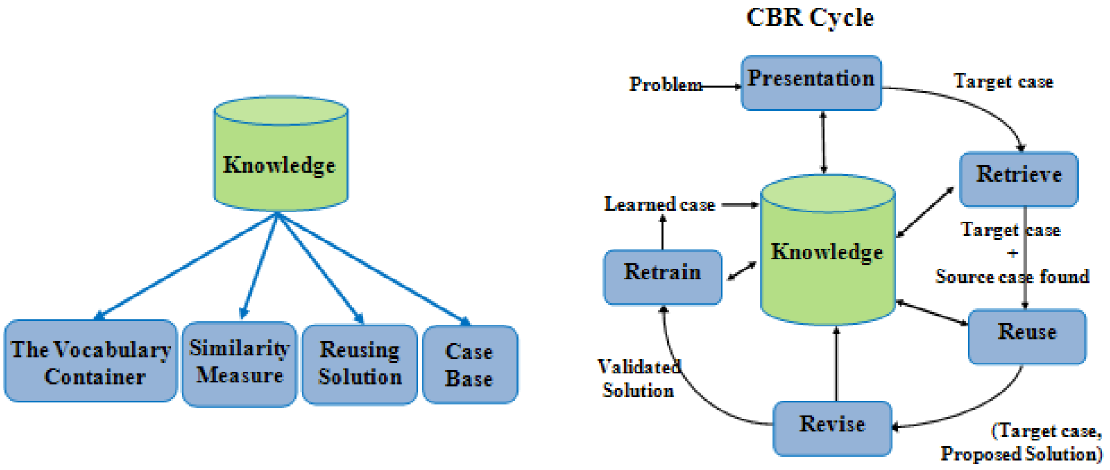

Fig. 1 The CBR components (Source [1], [12])

The systems based on the case-based reasoning can be classified into two categories of applications [18]:

- Applications for static situations. For this type of system, the designer must have all the characteristics describing a case, in advance, in order to be able to realize its model. A data model of the field is thus refined through an expertise in the field of application which can characterize a given situation. Thus, the cases are completely structured in this data model and often represented in a list (a: attributes, v: values). For example we have the system CHIEF [13] case base planner that builds new plans out of its memory of old ones. We do not exploit this type of CBR to develop our system, because in the approach for static situation, a problem must be completely described before starting the first steps. Nevertheless in our situation, the learner traces (target case) evolve dynamically over time, so we must treat a dynamic situation with some particular features.
- Applications for dynamic situations. They differ when we compare them to static cases by the fact that they deal with temporal target cases (the situation), by looking for similar cases (better cases) based on a resemblance between histories (for more details on the subject, the reader may refer to [2], [18])). Several works relate to dynamic case based reasoning such as REBECAS [18] prediction of processes from observed behaviors, application to wildfire and SAPED [2].

V. MULTI-AGENT CASE-BASED REASONING

The Multi-Agent Systems based on case based reasoning are used in many applications areas [25]. We can distinguish two types of applications (Table 5.1):

- The Multi-Agent Systems in which each agent uses the case based reasoning internally for their own needs (level agent case based reasoning): This type is the first model that was applied in Multi-Agent CBR Systems. For this type of system, each agent is able to find similar cases to the target case in their own case base, also able to accomplish all steps of CBR cycle. For example we have the system POMAESS in e-service field [32], CCBR framework to personalized route planning [21], and MCBR [17] for distributed systems.
- The Multi-Agent Systems whose approach is a case based reasoning (level Multi-Agent Case Based Reasoning) : For this types of applications, the Multi-Agent Case Based Reasoning System distribute the some/all steps of the CBR cycle (Representation, Retrieve, Reuse, Revise, Retain) among several agents. The second category of applications might be better than the first. Indeed the individual agents experience may be limited, therefore their
- knowledge and predictions too, so the agents are able to cooperate with other agents for a better prediction of the situation and they can benefit from the other agents capabilities. For example we have PROCLAIM [29] in argumentation field, and CBR-TEAM [26] approach that uses a set of heterogeneous cooperative agents in a parametric design task (steam-condenser component design).

TABLE I
CBR SYSTEM BASED MAS

CBR-MAS	CBR-Agent	Cycle
CBR-TEAM[26], MCBR [17], ProCLAIM[29]	CCBR[21], RoBoCats[20], AMAL [27], POMAESS [32], S-MAS [24]	Static

To our knowledge, no dynamic CBR Cycle reasoning system exists. The following table (Table 5.2) shows a summary of the CBR systems from static versus dynamic target case and static CBR cycle.

TABLE III

CBR SYSTEMS CLASSIFICATION

Target case	**CBR Cycle**	**Classical CBR Systems**	**CBR-MAS**	**CBR-Agent**
Static	Static	CHEF[13], CREEK [3], CASEY [15], RADIX [8]	CCBR[21], AMAL [27]	ProCLAIM[29]
Dynamic	Static	REBECAS [18], AuRA [16] , SAPED[2], CASEP2 [33], SBR[4]	CICLMAN [28], RoBoCats [20] , POMAESS [32], S-MAS [24]	MCBR [17], CBR-TEAM[26]
Dynamic	Dynamic			IDCBR-MAS (Our approach)

We propose a system called Dynamic Case Based Reasoning-Multi-Agent Sys-tem (DCBR-MAS), able to find similar cases to the target case in their own case base. Our system is founded on 1) a dynamic cycle of case-based reasoning, and 2) a dynamic retrieving method based on a complementary similarity measure, named Inverse Longest Common Sub-Sequence (ILCSS) (for more details on the subject, the reader may refer to [10, 34]).

VI. INCREMENTAL DYNAMIC CASE-BASED REASONING FOUNDED ON MULTI-AGENT SYSTEM

Our problem is similar to the CBR for dynamic and incremental situations. Indeed, the traces left by the learner during the learning session evolve dynamically over time; the case-based reasoning must take into account this evolution in an incremental way. In other words, we do not consider each evolution of the traces as a new target. The intelligent system (IDCBR-MAS) which we propose offer important features:

- It is dynamic. Indeed we must continually acquire new knowledge to better reproduce human behavior in each situation.
- It is incremental, this is its major feature because the trace evolves in a dynamic way for the same target case.

The main benefits of our approach are the distributed capabilities of the Multi-Agent Systems and the self-adaption ability to the changes that occur in each situation. The system we propose consists of the three layers components (as indicated in Figure Fig. 2:

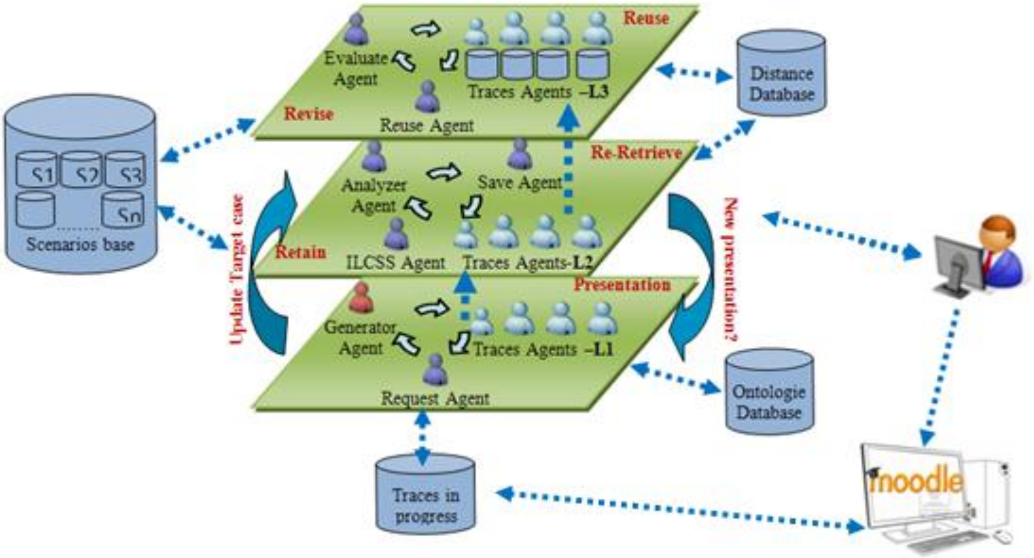

Fig. 2 General architecture of our approach.

A. Presentation layer

The role of agents of this layer is to manage information arrived from the environment (the learner traces). This information (the learner traces) feed the representation layer. The goal of this layer is to be both, a picture of the current situation being analyzed and to represent the dynamics of its evolutions over time. The target case, of the dynamic and incremental case based reasoning, is developed by this layer. The presentation layer contains the following agents:

- Request Agent: The role of this agent is to establish the link between the system and the environment. They feed the system with information from Distributed Information Systems (file traces). Also the goal of this agent is to check if there is any change in the traces file.

- Generator Agent: the role of this agent is created and/or updated TracesAgents-L1: The Request Agent transmits the data received from environment to Generator Agent. Two cases of figure are presented : if the Traces Agents-L1 (i) related to the learner i exists then this last will be updated, else the Generator Agent create a new Traces Agents-L1 (i) able to represent the learner i.
- TracesAgents-L1: For each Lerner i we have a represented TraceAgent-L1(i).These agents will encapsulate the original traces of learners.

B. Interpretation Layer and storage.

A set of agents allows the comparison between the current situation and past situations stored in the memory system (scenarios). In our approach the evaluation of the similarity between the current situation and similar past situations is process continues. In fact, the change of the current situation (due to the arrival of new information from the traces file) can change the list of similar past experiences found in the previous step (past experiences judge similar to the current situation). So the retrieve steps of CBR cycle of our system must take into account the change in the current situation. Our system will be able to repeat the retrieve steps following the change of the current situation or whenever necessary (retrieve scenarios or past experiences that are similar to the target case (the situation)). In addition, in our system the sequence of steps of the CBR cycle isn't important: in fact our system can stop each phase of the CBR cycle and return to a previous step following the change of the current situation or whenever necessary, and the order (Figure Fig. 3) presentation – retrieve – reuse – revise – retain is not static or fixed, it can change and some steps can be re-run each time until the change in our situation. The agents of this layer store and manage the new situation (new scenario).

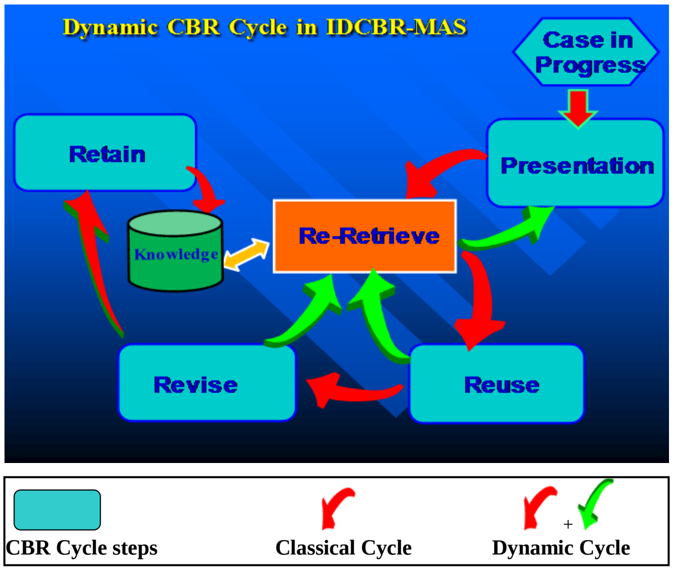

Fig. 3 Dynamic CBR Cycle in IDCBR-MAS

The Interpretation Layer contains the following agents:

- Traces Agents- L2: These agents contain the same information and data that have in Trace Agents-L1 of the first layer. They differ by an abstraction of the data, originally described and managed by Trace Agents-L1, that make it comparable to the scenarios (past experiences) stored in the memory.
- ILCSS Agent: The role of this agent is to evaluate in a continuously way the similarity between the current situation and scenarios stored in the memory based on the similarity measure ILCSS. The retrieve steps of our system is based this agent. It is one important step within the case-based reasoning paradigm. The success of retrieval step will depend on three factors: the case representation, case memory and similarity measure used to retrieve scenarios that are similar to the target case (the situation). ILCSS Agent stores the distances between the current situation and past scenarios in Distance Table. It is responsible for reviewing these distances every time following the change of the current situation or whenever necessary. For more details on similarity measure Inverse Longest Common Sub-Sequence (ILCSS), the reader may refer to [10]).
- Analyzer Agent: The goal of this agent is to check in a dynamic way if there is any change or update in Trace Agent -L2 (with the arrival of new information and data from the environment), then the Analyzer Agent asks ILCSS Agent to update Distance Table each time they have a change in Trace Agent -L2, if not they asks Request Agent of first Layer if there is any change in traces file.

C. Prediction Layer

The role of agents of this Layer is to predict the current situation by reusing past experiences selected by second Layer. The choice of similar paste experiences is evaluated by this layer, so one of these scenarios will be proposed to the learner. For each particular situation, the prediction agents can react differently depending on the learner profile concerned, for example, decided to initiate a communication session with a learner in difficulty. The Layer contains the following agents:

- Traces Agents-L3: At this stage of reasoning the system adds a pointer to each agent TracesAgents-L2. So the Traces Agents-L3 is identical to Traces Agents-L2 with a small difference, in fact for each Traces Agents-L2 we associate a list of similar scenarios through a pointer to the list of similar past experiences . The advantage of a pointer is that the list is not exhaustive and it changes dynamically over time following the change of the learner traces.
- Reuse Agent: The role of this agent is to predict future events of the situation by reusing the past experiences to the current situation.
- Evaluate Agent: The role of this agent is to evaluate the proposed solution by Reuse Agent and to ensure that the similarity between the current situation and scenarios chosen by the Prediction layer is sufficient.
- Human Tutor or Human Agent: The human tutor is solicited if the system detects a learning situation requiring his intervention (failure to find one or more similar scenarios to the current situation). It represents the human expert.

VII. IDCBR-MAS SYSTEM MODELLING

Our system IDCBR-MAS is composed of multiple interacting intelligent agents; it supports the specification, analysis, design and validation of our systems. We present the sequence diagram to assemble the various interactions carried out between the various actors of the platform.

The presentation of the situation (learner's traces) by the platform is a task managed by the several agents of the presentation layer of our system IDCBR-MAS, in addition, these agents are responsible for the update of the traces. The following sequence diagram illustrates the process of the situation presentation of the learner's traces.

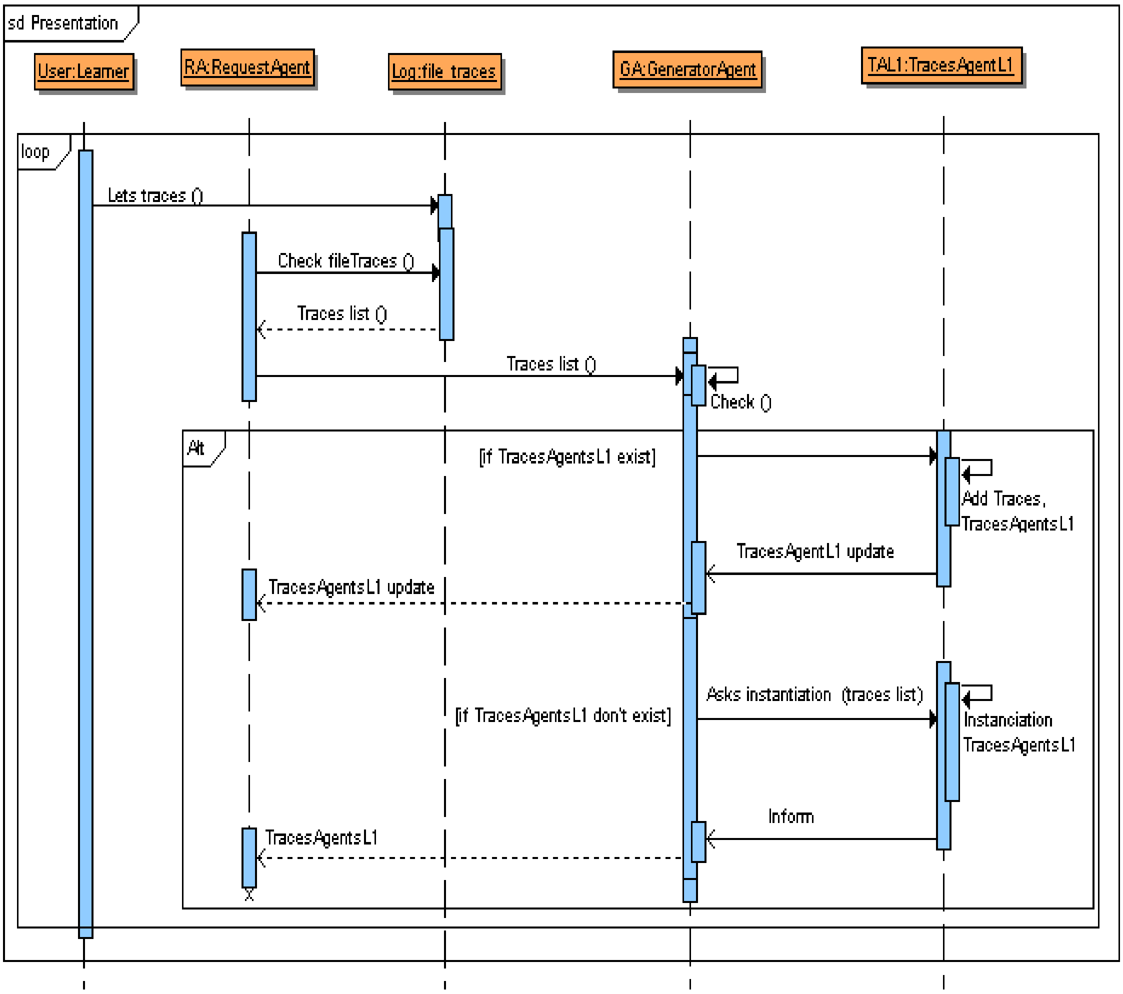

Fig. 4 The sequence diagram of the case presentation in system IDCBR-MAS.

Firstly the Request Agent addresses a request to server in order to retrieves the learner's traces left by the learner during the learning session and sending it to the Generator Agent, this last created/update the TracesAgents-L1: Two cases of figure are presented during the checking, if the Traces Agents-L1 (i) related to the learner i exists then the Traces Agents-L1 (i) will be updated, else the Generator Agent create a new Traces Agents-L1 (i) able to represent the learner i. the process will be re-run each time there is a change in the learner's traces :the agent check periodically for any change in the learner's traces and then the process will be re-executed.

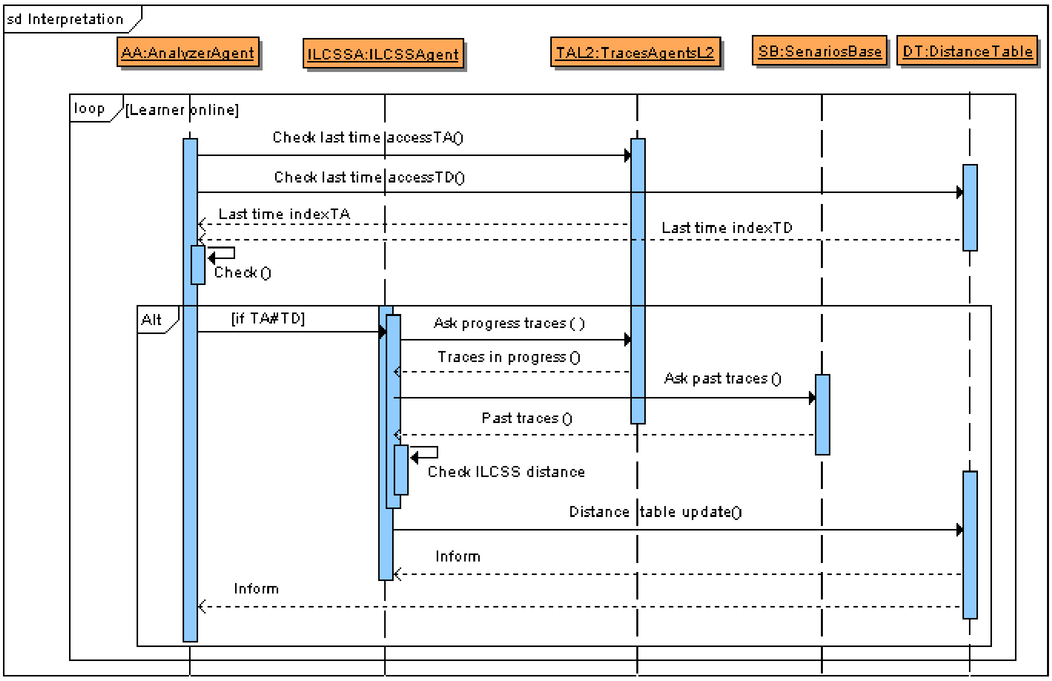

Fig. 5 The sequence diagram of the case Interpretation in system IDCBR-MAS

Firstly the Analyzer Agent (AA) addresses a request to Traces-Agents-L2 and to DistanceTable in order to retrieves two chronological dates TA: the date of last updated in the file traces and DT: the date of last update of distance table. The Analyzer agent check If TA= DT. If the two dates are not equal then the AA ask the ILCSSAgent to update the distance table which contains the distance between the current situation Traces-Agents-L2 and the scenario stored in memory. This is based on the similarity measures ILCSS. The agent also asks periodically the RequestAgent if there is any change in the learner's traces, whether the process will be re-executed.

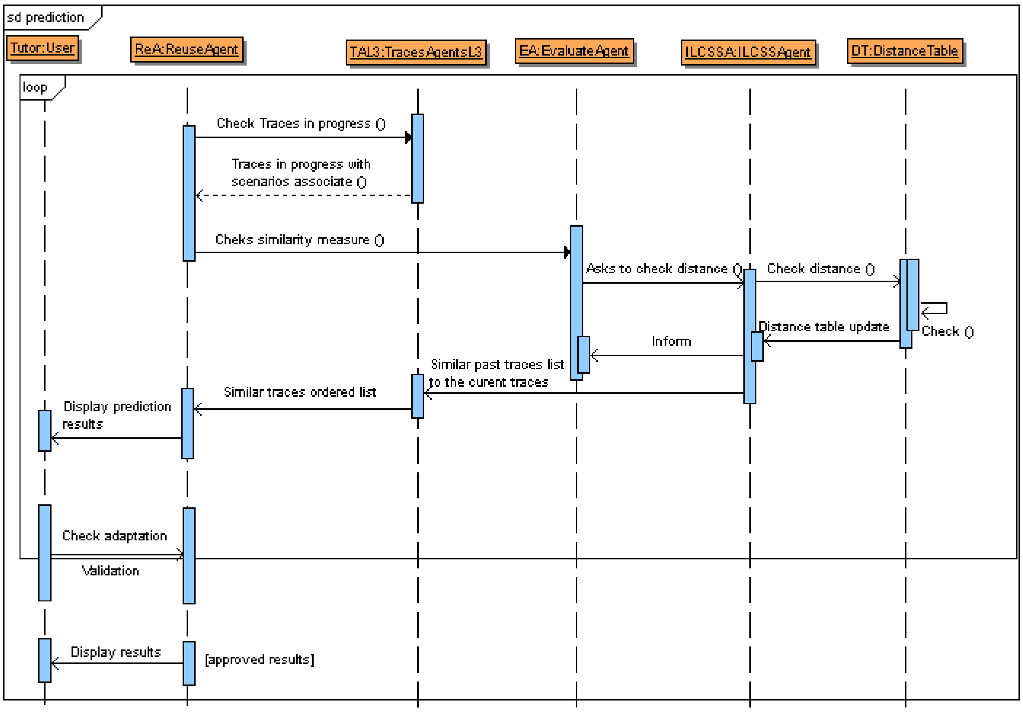
sd prediction
Tutor:User
ReA:ReuseAgent
TAL3:TracesAgentsL3
EA:EvaluateAgent
ILCSSA:ILCSSAgent
DT:DistanceTable
loop
Check Traces in progress ()
Traces in progress with scenarios associate ()
Cheks similarity measure ()
Asks to check distance ()
Check distance ()
Check ()
Distance table update
Inform
Similar past traces list to the curent traces
Similar traces ordered list
Display prediction results
Check adaptation
Validation
Display results
[approved results]

Fig. 6 The sequence diagram of the case prediction in system IDCBR-MAS

First of all the Reuse Agent ask the Traces-Agents-L3 to retrieve the current traces with the associated scenarios (the associated scenarios to the current traces are the scenarios that are very similar at learner's traces or target, based on the similarity measures ILCSS). Then the Evaluate Agent check the distance table. If necessary the Reuse Agent asks the ILCSS Agent asks to check and update all distances between the current situation (target case or learner's traces) and scenarios stored in memory and. Finally there will be a list of scenarios to proposed as for the situation prediction

VIII. CONCLUSION AND FUTURE WORK

Our system allows connecting and comparing the scenario found (current situation) to past scenarios that are stored in a database. The continuous analysis of information coming from the environment (learners traces) makes it possible to suggest to various actors (learners and tutor) possible evolutions of the current situation. The Multi-Agent architecture that we propose is based on three layers of agents with a pyramidal relation. The lower layer allows building a representation of the target case, i.e. the current situation. The second layer implements a dynamic process of the source cases recall allowing the search for past situations similar to the current one. Finally, the prediction layer captures the responses sent by the second layer to transform them into actions proposed either by machine tutor, virtual tutor, or/and human tutor.

We have presented systems founded of Incremental and Dynamic Case Based Reasoning and we have also clarified that the CBR-based applications can be classified according to the study area: CBR for static situations and CBR for dynamic situations. In our situation, we have used a Dynamic system IDCBR, which the second step of our IDCBR cycle is dynamic in order to push the limits of CBR cycle static. In fact, the current situation (target case) is a trace that evolves; the case based reasoning must take into account this evolution incrementally. In other words, it shouldn't consider each evolution of the trace as a new target case. Our future work consists in realizing a real experiment in a platform e-learning and a comparative study between our system and other tools.

REFERENCES

[1] A. Aamodt and E. Plaza, , Case-Based Reasoning : Foundational Issues, Methodological Variations, and System Approaches , AI Communications, 7(i), 1994.

[2] A. Aich, Reconnaissance et prdiction de situations dynamiques : application l’assistance de personnes handicapes moteur, Thse de doctorat, Universit de Technologie de Troyes, 2007.

[3] A. Aamodt, Knowledge-Intensive Case-Based Reasoning and Sustained Learning. Proc. of the 9th European Conference on Artificial Intelligence, ECCBR04, Lecture Notes in Artificial Intelligence, pp.1-15, Springer, 2004.

[4] C. Baptiste, Vers la gestion de scnarios pour la reconnaissance et l´anticipation de situations dynamiques , Universit de Technologie de Troyes, France, 2011.

[5] P. Brusilovski, Distributed Intelligent Tutoring on the Web. 8thWorld Conference of Artificial Intelligence in Education. IOS Press. pp.482-489, 1997.

[6] A. Cordier, B. Mascret, A. Mille, Dynamic Case Based Reasoning for Contextual Reuse of Experience, Case-Based Reasoning Workshop, ICCBR 2010, Cindy Marling ed. Alessan-dria, Italy. pp. 69-78. 2010.

[7] G. Constantino, D. Suthers and J-I. Icaza, Designing and Evaluating a Collaboration Coach: Knowledge and Reasoning, Proceedings of the Artificial Intelligence in Education, AI-ED, J.D. Moore et al (Eds). The Netherlands: IOS Press. pp. 176-187, 2001.

[8] F. Corvaisier, Mille A., Pinon J.M., Radix 2, assistance la recherche d’information documentaire sur le web, In IC’98, Ingnierie des Connaissances, Pont--Mousson, France, INRIALORIA, Nancy, 1998, p. 153-163.

[9] M. Ennaji, H. Boukachour, P. Grav, Une architecture Multi-Agent pour la pedagogie de la formation distance, In: MOSIM 2006, Rabat, Maroc (2006).

[10] E. M. En-Naimi, A. Zouhair, B. Amami, H. Boukachour, P. Person and C. Bertelle , Intelligent Tutoring Systems Based on the Multi-Agent Systems (ITS-MAS): The Dynamic and Incremental Case Based Reasoning (DICBR) Paradigm and the Inverse Longest Common Sub-Sequence (ILCSS) in the CEHL, The IJCSI, Vol. 9, Issue 6, No 3, November 2012.

[11] C. Frasson, L. Martin, G. Gouarderes and E. Ameur, A distance learning Architecture Based on Networked Cognitive Agents, Intelligent Tutoring Systems, 1998.

[12] B. Fuchs, J. Lieber, A. Mille and A. Napoli, , Une première formalisation de la phase d'élaboration du raisonnement partir de cas, Actes du 14ime atelier du raisonnement partir de cas, Besanon, mars, 2006.

[13] K.J. Hammond, CHEF: a model of case-based planning, Proc. Of AAAI86, Morgan Kaufman, pp. 267-271, 1986.

[14] J. Kolodner, , Case-Based Reasoning, Morgan Kaufmann, San Mateo, UCA, 1993.

[15] P. Koton, Using experience in learning and problem solving, PhD Thesis, Laboratory of Computer Science, Massachusetts Instite of Technology, Cambridge Massachusetts, USA 1988.

[16] M. Likhachev, M. Kaess, and R.C. Arkin, Learning behavioral parameterization using spatiotemporal case-based reasoning, The IEEE international conference on robotics and automation (ICRA02), Washington, D.C, 2002.

[17] D. Leake and R. Sooriamurthi, When two case bases are better than one: Exploiting multiple case bases, In ICCBR, pages 321335, 2001.

[18] S. Loriette-Rougegrez, , Raisonnement partir de cas pour des solutions spatiotemporelles de processus, revue internationale de gomatique, vol 8, n 1-2, 1998.

[19] J-C. Marty and A. Mille, , Analyse de traces et personnalisations des environnements informatiques pour lapprentissage humain, Edition Lavoisier, 2009.

[20] C. Marling, M. Tomko, M. Gillen, D. Alexander, and D. Chelberg, Case-Based Reasoning for Planning and World Modeling in the RoboCup Small Size League, School of Electrical Engineering and Computer Science, Ohio University, Athens, Ohio 45701, USA.

[21] L. McGinty and B. smyth, Collaborative case-based reasoning: Applications in personalized route planning, In ICCBR, pages 362376, 2001.

[22] A. Mille, , From case-based reasoning to traces-based reasoning, Annual Reviews in Control30(2):223-232, ELSEVIER, ISSN 1367-5788. 2006.

[23] J. Piaget, Psychologie et pédagogie, Paris: Denol-Gonthier, 1969.

[24] I. Pinzn Cristian, B. Javier, F. Juan, M. Juan, B. Corchado, S-MAS: An adaptive hierarchical distributed multi-agent architecture for blocking malicious SOAP messages within Web Services environments, Expert Systems with Applications Volume 38, Issue 5, May 2011, Pages 5486-5499, http://www.sciencedirect.com/science/article/pii/S0957417410012327

[25] E. Plaza and L. Mcginty, , Distributed case-based reasoning, The Knowledge Engineering, Review, Vol. 00:0, 14.c 2005, Cambridge University Press, 2005.

[26] M. Prassad, V. Lesser, and S. Lander, Retrieval and reasoning in distributed case bases. Technical report, UMass Computer Science Department, 1995.

[27] S. Ontan and E. Plaza, Learning and Joint Deliberation through Argumentation in Multi-Agents Systems, in AAMAS'07 Proceedings of the 6th international joint conference on Autonomous agents and Multi-Agents systems, Honolulu, HI, USA, 2007

[28] L. k. Soh and J. Luo, Combining individual and cooperative learning for multiagent negotiations, In Int. Conf. Autonomous Agents and Multiagent Systems AAMAS03, pages 686692,2003.

[29] P. Tolchinsky, S. Modgil, U. Cortes, and M. Sanchez-marre, Cbr and argument schemes for collaborative decision making, In Conference on computational models of argument, COMMA-06 (Vol. 144, pp. 7182), 2006.

[30] Vygotski L-S. , Mind in society: the development of higher psychological processes, Harvard University Press, Cambridge, MA, 1978.

[31] C. Webber and S. Pesty, , Emergence de diagnostic par formation de coalitions - Application au diagnostic des conceptions d'un apprenant, Journes Francophones pour l'Intelligence Artificielle Distribue et les Systmes Multi-Agents, Hermes, Lille, pp.45-57, 2002.

[32] R. Yua, B. Iunga, H. Panetto, A multi-agents based E-maintenance system with case-based reasoning decision support, Faculte des Sciences, CRAN-Universite de Nancy I, and Faculty of Hydroelectric Power, Huazhong University of Science and Technology, Wuhan, China, in Engineering Applications of Artificial Intelligence 16 (2003) 321333.

[33] F. Zehraoui, Systèmes d'apprentissage connexionnistes et raisonnement partir de cas pour la classification et le classement de séquences, Thèse de doctorat, Université Paris13, 2004.

[34] Abdelhamid ZOUHAIR, El Mokhtar EN-NAIMI, Benaissa AMAMI, Hadhoum BOUKACHOUR, Patrick PERSON, Cyrille BERTELLE, Intelligent Tutoring Systems Founded on The Multi-Agent Incremental Dynamic Case Based Reasoning, IEEE CIST'12, The 2nd IEEE international Colloquium in Information Science and Technology, Fez, Morocco, October 2012.