

AI-VT: An example of CBR that generates a variety of solutions to the same problem

Julien Henriet, Françoise Greffier

► **To cite this version:**

Julien Henriet, Françoise Greffier. AI-VT: An example of CBR that generates a variety of solutions to the same problem. ICCBR 2018 – 26th International Conference on Case-Based Reasoning, July 9-12 in Stockholm, Sweden, Jul 2018, Stockholm, Norway. 10.1007/978-3-030-01081-2_9. hal-02103470

HAL Id: hal-02103470

<https://hal-univ-fcomte.archives-ouvertes.fr/hal-02103470>

Submitted on 18 Apr 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.



AI-VT: An example of CBR that generates a variety of solutions to the same problem

Julien Henriet¹ and Françoise Greffier²

¹ FEMTO-ST Institute
Univ. Bourgogne-Franche-Comté, CNRS
DISC, 16 route de Gray, 25030 Besançon, France
julien.henriet@univ-fcomte.fr

² ELLIADD
Univ. Bourgogne-Franche-Comté
16 route de Gray, 25030 Besançon, France
francoise.greffier@univ-fcomte.fr

Abstract. AI-Virtual Trainer (AI-VT) is an Intelligent tutoring system based on Case-based reasoning. AI-VT has been design to generate personalised, varied and consistent training sessions to learners. It has been thought as a system which can help learners to train themselves to different abilities during a cycle. The training sessions of AI-VT propose different exercises in regards of a capacity. AI-VT can elaborate a list of exercises personalised to each learner. One of the main requirement and challenge studied in this work is its ability to propose varied training sessions to the same learner for many weeks which constituted the challenge studied in the present work. Indeed, if the same set of exercises is proposed time after time to learners, it will not keep their attention and motivation. Thus, even if the generation of training sessions is based on analogy and must integrate the repetition of some exercises, it also must introduce some diversity and AI-VT must deals with diversity. In this paper, we have highlighted the fact that the capitalisation phase of CBR is of the utmost importance for diversity, and we have also highlighted that the tension between reptition and variety depends on the abilities learned. This tension has an important impact on the capitalisation phase of AI-VT.

Keywords: Case-Based Reasoning · Intelligent Tutoring System · diversity · capitalisation · personalised learning

1 Introduction

We are interested in the issue of the personalisation of learning through training sessions. For us, a training session is a list of exercises suited to each learner. Motivation and repetitivity are key aspects in teaching. Nevertheless, repetitivity causes learners to be bored and to turn themselves off. Consequently, teachers must introduce originality and diversity, and adapt the exercise level and nature to the learners' already acquired skills. This is particularly true in

training sessions, where teachers must propose varied exercises and consistent sessions while having to train for the same skill over a given number of weeks. Thus, the elaboration of a training session of a cycle and suited to one particular learner is a reasoning based on analogy in which it is necessary to introduce some kind of originality. Indeed, on the one hand, this elaboration is based on the past experiences of the trainer as well as the exercises previously proposed to the learner, and on the other hand, the exercises proposed to the learner must not be always the same. As a consequence, a Case-based reasoning (CBR) system [10], based on analogy reasoning is a good candidate as an answer to that kind of systems, but must be adapted in order to introduce a kind of diversity in the solutions to be proposed (the training sessions). In addition, this diversity varies from a domain to another. Indeed, the frequency with which an exercise must be proposed to sports learners is not the same as learners who learn mathematics for example. As a matter of fact, basic exercises will be often proposed by sports trainers since the body must practise a lot before integrating basic movements and attitudes. On the contrary, if a learner has succeeded a math exercise, the fact of proposing it once again will be bored.

2 Related works

This paper presents Artificial Intelligent - Virtual Trainer (AI-VT), a Multi-Agent System (MAS) that uses CBR to provide consistent training sessions with widely differing progressions. CBR is widely employed in e-learning systems and Intelligent Tutoring Systems (ITS) [7]. J. L. Kolodner [11] distinguished between two types of CBR-inspired approaches to education: Goal-Based Scenarios [16] where learners achieve missions in simulated worlds thus confronting themselves with the real world, and Learning By Design [12] in which learners design and build working devices to obtain feedback. CBR is actually well-suited to the latter type of system [9], as well as to other tools from Artificial Intelligence (AI) and Distributed AI (DAI) systems such as Genetic Algorithm (GA) [2], Artificial Neural Network (ANN) [4] and MAS [17]. A. Baylari and G. A. Montazer focused on the adaptation of tests to obtain a personalised estimation of a learner's level [4]. They used an ANN in order to correlate learner answers to the tests and the exercises proposed by teachers. The CBR and GA based e-learning system proposed by Huang *et al.* also provides lessons taking into account the curriculum and the incorrect response patterns of a pre-test given to the learner [8]. O. P. Rishi *et al.* designed an ITS based on agents and a CBR system [15] in which a Personal Agent is responsible for determining learner level. A Teaching Agent then determines the educative strategy with the help of CBR regarding the description of the transmitted learner level. Finally, a Course Agent provides and revises the lessons and exercises corresponding to the strategy proposed by the system with the help of a tutor. All these tools provided by AI, would nevertheless produce exactly the same exercises and lessons to train for a single given skill, or would propose a large set of exercises as an answer to the diversity constraint, and leave the teachers or the learners to choose the

most adapted exercises themselves. In this particular domain, repetitive activities are a drawback, yet lesson planning is a process based on adaptation of past experiences.

AI-VT tries to adress the problem of tension between repetitivity and variety of the solutions consecutively proposed. Indeed, even if the exercices must be selected by analogy with past proposed ones, the same exercise proposed too often to one learner may bore her/him. Moreover, the number of propositions varies according to the domain (maths vs sports for example) and also the level reached by the learner. The problem of variety in CBR-system is close to the creativity one adressed in the litterature [13, 6, 5, 3, 14]. G. Muller and R. Bergmann proposed to introduce novelty combining different solutions during the adaptation phase [14]. In their approach, source case solutions are decomposed in elementary activities and elements and combined in an original way. This approach allows to introduce diversity and novelty in the solutions proposed by their system. J. Gero and M.L. Maher put the basis of a new approach based on Deep learning in order to introcuce creativity [3]. K. Grace *et al.* went further with Deep learning and proposed creative and unexpected concepts then adapted in a CBR-cycle process in order to generate original recipes [6, 5]. This neural network is trained to introduce novelty (new ingredients) into a set of preferences of the end-user in order to give receipes with new ingredients [13]. Actually, in this approaches, the creativity and originality are treated during the description and adaptation phases of the target case whearas AI-VT adresses this particular aspect during the capitalisation and adaptation phases, giving much importance to these phases of this CBR-system.

3 Presentation of AI-VT

AI-VT is based on a pedagogy based on capacities and sub-capacities to be integrated. Sub-capacities constitute the steps to reach. They can be linearly organised. Once a lack of knowledge is detected by the teacher or the learner, she/he can decide to train for many weeks in order to reach this step. Then, when the user asks for a training session on a particular capacity to AI-VT, the system generates a session organised in sub-capacities and proposes exercices in regards of each sub-capacity. In the first part of this section, we detail the session structure and the requirements of AI-VT. The distributed architecture and the data flows are presented in the second part. Finally, in the third part, we examine how a session is designed.

3.1 Lesson structure

In this sub-section, we describe the way a teacher elaborates a training session, the parameters and the way this generation is done, and the behavior AI-VT should imitate.

We considered activities for which each training session is guided by reaching one capacity [18]. These capacities and their order of appearance are decided at

the beginning of each session. One specific skill can consequently be assigned to some consecutive sessions. The chosen capacity is then divided into elementary abilities (sub-capacities) that have to be mastered by the learner. We considered that in sports in particular, each skill may be shared by more than one capacity. In all the domain of application we considered, the mastery of each skill is a time consuming process that is reached through the repetition of exercises [1]. The delay required for mastering a single skill depends on the learner. Some will learn faster than others, and thus the teacher must adapt each session to the level of the learner. Consequently, a session is a sorted set of exercises carefully chosen by the teacher. Some exercises are more complicated or fundamental than others. Ideally, a session begins with simple exercises that prepare the learner for the next ones. This implies that a certain complexity must be associated with each exercise and that a measure of the differences between exercises must be taken into account. Thus, we introduced a notion of distance between exercises so that AI-VT could measure the differences between two consecutive exercises proposed in each session. This computation of distance will help the system to stay in the same theme of learning.

Consequently, AI-VT must:

- (i) propose pertinent sub-capacities and exercises in function of the capacity decided and the level already reached by the learner,
- (ii) ensure that no exercise is proposed more than once during a given training session and that the sessions in the same training cycle are varied,
- (iii) build a consistent training session such that it begins with the simplest exercise to then continue with a list of exercises sufficiently related to the preceding and following ones.

3.2 System architecture and communication model

Figure 1 presents the architecture of AI-VT modeled as a Multi-Agent System (MAS).

MAS constitute a paradigm designed to handle distributed systems. In an MAS, an agent is a physical or abstract entity having certain specific characteristics: perception of its environment (including itself and the other agents), the capability to act (upon itself or upon the environment) and autonomy in its decisions and actions.

In AI-VT, the choice of sub-capacities regarding a given capacity takes place via an autonomous process, as does the determination of exercises regarding a sub-capacity, or of any other exercises chosen and their priority levels. The initial choice of exercises regarding a sub-capacity must be an autonomous process: each agent's autonomy ensures a wise and free selection of the most adapted exercises. These processes can be undertaken simultaneously, coming after the determination of sub-capacities. In addition, each one must interact with the other processes and takes their choices into account: the solution proposed by one agent influences the choices made by the others.

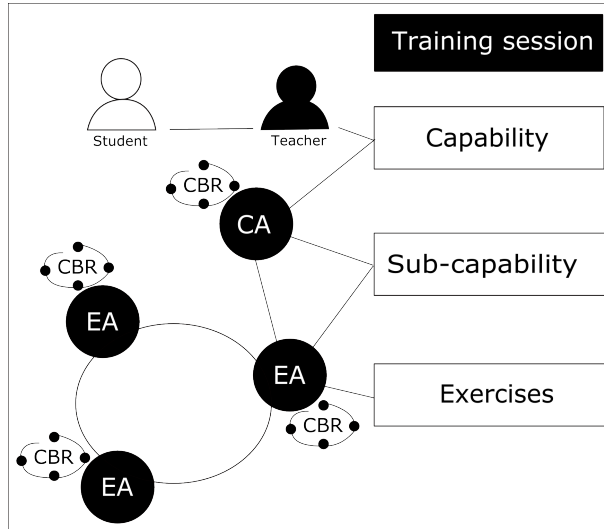


Fig. 1. Overview of AI-VT Architecture

As shown in Figure 1, the system is composed of four types of agents: the teacher, the learner, the Capacity Agent (CA) which is responsible for choosing the sub-capacities regarding a capacity requested by the teacher, and the exercise agents. Each of these agents is responsible for proposing the exercises best suited to a given sub-capacity. The CA sends the set of sub-capacities it has chosen to one of the exercise agents. This first-contacted Exercise Agent (EA) endorses the role of coordinator between the CA and the other EAs. This EA assumes responsibility for the first-proposed sub-capacity, and then creates and sends the list to another EA which assumes the second sub-capacity.

The EAs then communicate and share information in order to prepare the requested training session. Each EA proposes exercises concerning its assigned sub-capacity. Nevertheless, during this process, each EA must take into account the choices proposed by the other EAs: for example, one of the system's constraints is that each exercise is to be done only once during the entire training session. Thus, the choices of the EAs are shared. The referent version of the training session is transmitted from EA to EA until it answers to all the requirements. Finally, the EA initially contacted by the CA sends the referent version of the training session back to the CA.

3.3 Determination of sub-capacities

The CA is responsible for choosing the set of sub-capacities and their duration. Once the training session has been chosen by the teacher, and after having analysed any additional learner needs, the CA follows the CBR approach to make these choices in function of the sub-capacities already achieved and according to the learners' degree of assimilation.

For the CA, a case is a set comprised of two parts: a problem and a solution. Each problem part is composed of a capacity C , and the solution part of a set of $(SC, D_{SC,C})$ where SC is a sub-capacity and $D_{SC,C}$ the duration required to reach this SC regarding C . Thus, formally, a source case s is expressed as $s = (C, \cup\{SC, D_{SC,C}^S\})$. The durations $D_{SC,C}$ are initialised by the teacher at the beginning of the season and updated by AI-VT after the training session, taking into account the remarks made by the teacher. Since learner levels of expertise rise, we can consider that durations decrease and thus call them 'remaining durations'.

First, all the sub-capacities associated with C are retrieved. The similarity between each source case s and the target case t is computed as follows:

$$SIM_{SC}(t, s) = \begin{cases} 1 & \text{if } C_t = C_s \\ 0 & \text{if } C_t \neq C_s \end{cases} \text{ where } C_s \text{ (respectively } C_t) \text{ is the capacity of } s \text{ (respectively } t).$$

In order to illustrate this phase, we can consider the cases stored in the case base reported in Table 1. In this example taken from an Aikido training session, sub-capacities 'Breaking the partner's posture', 'Relaxing despite a grip', 'Making the partner lose balance' and 'Pivoting around a grip' have been associated with the capacity 'Using a grip' by the trainer in the initial process or in previous training sessions. Thus, if the capacity of t is $C_t = \text{'Using a grip'}$, the sub-capacities of source case 1 are reminded by analogy, since $SIM_{SC}(t, 1) = 1$ and $SIM_{SC}(t, 2) = 0$.

The adaptation phase consists of computing the duration of each sub-capacity. We assumed that these durations are somehow linked to the importance of practising each sub-capacity regarding the given capacity. Consequently, the adaptation module sorts the set of sub-capacities according to $D_{SC,C}$ (descending order). Then, the proposed durations are calculated according to the number of sub-capacities the teacher wants to work on.

Following the training session, the teacher evaluates the learner's acquired level of mastery. Before the training session begins, each selected sub-capacity is transmitted to one EA that will have to associate the corresponding exercises. After the training session, each sub-capacity duration is modified in proportion to the evaluation from 0 to 10 of the learner's level for the proposed sub-capacity.

Finally, during the capitalisation phase, the system subtracts the durations from all the durations of the source cases for which the practiced sub-capacity appear. During the capitalisation, the effective time spent by the learner to resolve the exercises of the sub-capacity may differ from the delay of resolution initially allocated by the teacher. Since this difference between time really spent and time initially allocated gives information on the difficulties felt by the learner and the integration of the sub-capacity, it has been taken into account into the remaining time to spend on the sub-capacity.

For that reason, the remaining duration of each worked sub-capacities is computed as follows:

$$D_{SC} = D_{SC} - \frac{M_{SC}}{10} \times \frac{d_{SC}^{alloc}}{d_{SC}^{real}},$$

where M_{SC} is the mark (out of 10) obtained by the learner, d_{SC}^{alloc} is the predicted

duration allowed to finish the exercises of the sub-capacity and d_{SC}^{real} is the real duration spent by the learner on the exercises of the sub-capacity.

Table 1. Example of modifications of durations after a training session.

Source case	Capacity	Sub-capacities	Initial duration (min.)	Teacher's mark (points)	Stored duration (min.)
1	Using a grip	Breaking the partner's posture	90	7/10	$90 - \frac{7}{10} \times \frac{20}{20} = 76$
		Relaxing despite a grip	90	3/10	$90 - \frac{3}{10} \times \frac{20}{20} = 84$
		Making the partner lose balance	80	4/10	$80 - \frac{4}{10} \times \frac{20}{20} = 72$
		Pivoting around a grip	80	-	80
2	Breaking	Breaking a single grip	90	-	90
	a grip	Relaxing despite a grip	70	3/10	$70 - \frac{3}{10} \times \frac{20}{20} = 64$

Table 1 presents two source cases of the tests performed by one of the trainers (an aikido teacher) who evaluated AI-VT. The trainer chose 3 different sub-capacities. Since the trainer chose the capacity *become capable of 'Using a grip'* for this training session, the CA recalled case 1. The adaptation process then sorted the sub-capacities according to their durations and proposed the three first sub-capacities, allocating $D_{SC} = \frac{60}{3} = 20$ minutes to each sub-capacity. In this example, the times really spent were all equals to the times allocated. It was very usual for sports training tests, whereas very unusual for algorithm trainings. Consequently, after capitalisation, the new durations were those reported in the last column of Table 1. Thus, the less assimilated sub-capacities (*'Relaxing despite a grip'* and *'Pivoting around a grip'*) became the most immediate ones. We also note that, as required for the system specification, when the same capacity (*'Using a grip'*) was selected again, another set of sub-capacities (composed of the less assimilated ones and others) were selected. Thus, as required, the proposed solutions changed even if the same capacity was requested again later.

3.4 Selection of varied exercises

This subsection presents how the exercises are chosen regarding the selected sub-capacities. At the allocation of its sub-capacity, each EA selects a set of exercises according to the CBR-cycle. For each EA, source case is noted:

$$\sigma = (SC, \bigcup\{(EX, AD_{EX}^\sigma, RD_{EX,SC}^\sigma)\}),$$

where AD_{EX}^σ is the estimated duration that must be allocated to the learner to resolve the exercise EX , and $RD_{EX,SC}^\sigma$, the estimated remaining duration to spend on this exercise EX before reaching the sub-capacity SC . Each source case σ contains the exercises possible regarding SC . Assuming $Card(Sol_\sigma)$ is the number of exercises of the solution part of σ , the target case τ_i (i.e. the part of the training session that will be proposed) taken into account by the EA EA_i is noted:

$$\tau_i = (SC_i, \bigcup_{n \in \{1..Card(Sol_\sigma)\}} \{(EX_n, AD_{EX_n}^{\tau_i}, RD_{EX_n,SC_i}^{\tau_i})\}).$$

Each EA_i then retrieves the source case corresponding to SC_i .

The similarity between source case σ and target case τ_i is computed as follows:

$$SIM_{EX}(\tau_i, \sigma) = \begin{cases} 1 & \text{if } SC_i = SC \\ 0 & \text{if } SC_i \neq SC \end{cases}$$

The adaptation phase orders the exercises of the training session. Selected exercises for which RD is the highest are proposed first. If two agents select the same exercise, the one with the highest RD_{EX,SC_i} prevails, and the one with the lowest must be changed. Then, exercises are ordered according to their complexity (ascending order). Finally, distances between consecutive exercises are computed and permutations between consecutive exercises may occur in order to minimise these distances between one proposed exercise and the next one. This final adaptation step creates a consistency for the training session.

During the revision phase, teacher and learner evaluate the answers (give a mark between 0 and 10) proposed by the learner and give the real duration spent on each exercise.

As an example, Tables 2 and 3 illustrate the adaptation of a training session dedicated to algorithm. Table 2 shows the sub-capacities retrieved by AI-VT. These sub-capacities are ordered by AI-VT according to their RD (descending order). If this training session was proposed to the learner, the context would change on each exercise (EX_6 and EX_2 deal with economy, EX_1 deals with geometry and EX_7 deals with fuel consumption). In addition, the more complex exercises would be proposed first, and the training would finish with the easiest exercise (EX_1).

Thus, at the end of its adaptation phase, AI-VT proposes a different order for the same exercises (the one in Table 3). In Table 3, we can see that the training session will begin with the easiest exercise (EX_1) and the two exercises that deal with economy are proposed one just after the other (EX_2 and EX_6).

At the end of the CBR-cycle, the capitalisation will allow the system to prepare the next training session. Indeed, even if the same sub-capacities are required next, the system will have to propose a different set of exercises. Thus,

Table 2. Example of exercises initially retrieved by AI-VT.

Sub-capacities / Exercises	Dist. with next ex.	Complexity	RD
- <i>Sub-capacity SC₃: Conception of an algorithm</i>			20
<i>EX₆:</i> Retrieve the total price before tax of product using knowing its price including tax and tax rate. Give the formula.	18	18	10
<i>EX₇:</i> Compute the fuel consumption of car knowing the distance and its mean speed. Give the formula.	18	18	10
- <i>Sub-capacity SC₁: Find inputs & outputs</i>			15
<i>EX₂:</i> Retrieve the total price before tax of product using knowing its price including tax and tax rate. Give the inputs & outputs.	18	18	10
<i>EX₁:</i> Compute the perimeter of a rectangle. Give the inputs & outputs.		5	5

Table 3. Example of exercises finally proposed by AI-VT.

Sub-capacities / Exercises	Dist. with next ex.	Complexity	RD
- <i>Sub-capacity SC₁: Find inputs & outputs</i>			15
<i>EX₁:</i> Compute the perimeter of a rectangle. Give the inputs & outputs.	18	5	5
<i>EX₂:</i> Retrieve the total price before tax of product using knowing its price including tax and tax rate. Give the inputs & outputs.	5	18	10
- <i>Sub-capacity SC₃: Conception of an algorithm</i>			20
<i>EX₆:</i> Retrieve the total price before tax of product using knowing its price including tax and tax rate. Give the formula.	18	18	10
<i>EX₇:</i> Compute the fuel consumption of car knowing the distance and its mean speed. Give the formula.		18	10

the capitalisation phase of AI-VT is much important since it will give the history of the worked exercises. Furthermore, if one exercise has not been understood or successfully solved, or even solved but with most difficulty by the learner, the system must have the possibility to chose this exercise again. Otherwise, if one exercise has been successfully solved with no difficulty, AI-VT must not propose it again... in the case of a theoretical knowledge acquisition.

Indeed, in the case of sports, for example, training sessions usually begin with most of the same basic exercises. Another difference between physical practice and theoretical acquisition resides in the fact that if an exercise has been correctly performed by an athlete, it can be proposed once again in the next training session.

In addition, if an exercise has been done with much difficulty by an athlete, the duration of practice may not change. On the contrary, for a theoretical training, if the learner has spent a lot of time on an exercise and did not solved it, this exercise should be proposed once again to the learner with a higher delay of resolution.

Consequently, AI-VT must capitalise differently cases of theoretical domain training and cases of physical training.

In the case of physical training, only the RD is modified as follows:

$$\forall SC, \forall EX, RD_{EX,SC}^{\sigma} = \max(0, (RD_{EX,SC}^{\sigma} - \frac{M_{EX}}{10} \times AD_{EX,SC}^{\tau})).$$

In that case, the remaining duration of practise of the exercises are only decreased from the time spent over it during the training session.

And in the case of training on theoretical skills, AD and RD are modified as follows:

$$\forall SC, \forall EX, AD_{EX,SC}^{\sigma} = \max(0, (AD_{EX,SC}^{\sigma} - \frac{M_{EX}}{10} \times \frac{AD_{EX,SC}^{Real}}{AD_{EX,SC}^{\tau}})),$$

$$\forall SC, \forall EX, RD_{EX,SC}^{\sigma} = \max(0, (RD_{EX,SC}^{\sigma} \times (1 - \frac{M_{EX}}{10} \times \frac{AD_{EX,SC}^{Real}}{AD_{EX,SC}^{\tau}}))),$$

where M_{EX} is the mark (out of 10) obtained by the learner for exercise EX and $AD_{EX,SC}^{Real}$ the real time spent over this exercise. For these kind of learning, we considered that the time spent by the learner over an exercise ($AD_{EX,SC}^{Real}$) can differ from the initial time allocated ($AD_{EX,SC}^{\tau}$).

In order to illustrate the ability of AI-VT to generate varied training sessions through its capitalisation phase, Table 4 presents the different durations (RD , AD and real time spent) of the exercises proposed in the last training session and the durations of other exercises stored in the case base. We can see that the capitalisation phase modifies the priorities of the exercises stored in the case base. Indeed, the RDs of the successfully resolved exercises fall to 0: EX_1 and EX_2 will not be proposed next time. In addition, since EX_6 has been partially resolved (mark 5/10) with high diffulty (time spent 12 min. instead of 8 min. planned), its RD becomes inferior to other exercises of the case-base: it could be proposed another time, but other exercises of the same sub-capacity will be selected first for the next training session. Finally, EX_7 has not been resolved at all (mark 0/10) and the learner has spend much time on it (15 min. instead of 8 min. planned). Consequently, its AD and RD stay the same and it will most probably be proposed next time with EX_8 .

Table 4. Example of capitalisation proposed by AI-VT.

Exercises	Initial <i>AD</i>	Initial <i>RD</i>	Mark	Real time spent	Capitalised <i>AD</i>	Capitalised <i>RD</i>
Ex. of SC_1 :						
EX_1 (proposed)	5	5	10	5	$5 - \frac{10}{10} \times \frac{5}{5} = 4$	$5 \times (1 - \frac{10}{10} \times \frac{5}{5}) = 0$
EX_2 (proposed)	5	10	10	5	$5 - \frac{10}{10} \times \frac{5}{5} = 4$	$10 \times (1 - \frac{10}{10} \times \frac{5}{5}) = 0$
EX_3 (case base)	5	5	-	-	5	5
EX_4 (case base)	5	5	-	-	5	5
Ex. of SC_3 :						
EX_6 (proposed)	8	10	5	12	$8 - \frac{5}{10} \times \frac{12}{8} = 7$	$10 \times (1 - \frac{5}{10} \times \frac{15}{8}) = 3$
EX_7 (proposed)	8	10	0	15	$8 - \frac{0}{10} \times \frac{15}{8} = 8$	$10 \times (1 - \frac{0}{10} \times \frac{15}{8}) = 10$
EX_8 (case base)	8	10	-	-	8	10

4 Results

AI-VT has been tested in two very different contexts. Its first context of use was sports training, and Aikido in particular, which is a traditional Japanese martial art. The second context of use was the training at algorithm, in the field of computer science.

4.1 Evaluation of AI-VT in the context of sports training (Aikido)

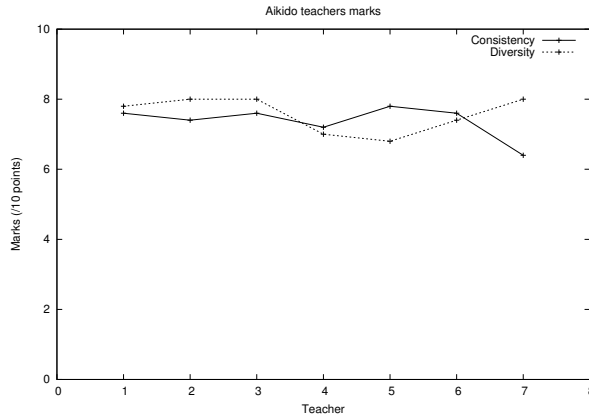


Fig. 2. Consistency and diversity marks obtained by the Aikido training sessions generated by AI-VT

We have asked 7 aikido teachers to evaluate 10 consecutive training sessions over the same capacity. This corresponds to 5 weeks training, with 2 sessions of 90 minutes per week. They evaluated the system through two aspects: the consistency of the proposed training sessions, and the diversity of the proposed exercises. It is important to notice that all the trainers had different sessions, and has initialised the system differently. Indeed, each trainer has its own way of learning Aikido, and a set of favorite techniques.

Figure 2 presents the evaluations of the Aikido trainers obtained by the training sessions generated by AI-VT. The mean marks obtained for each trainer are reported in this figure. The trainers were asked to give a mark out from 0 to 10 to the consistency of the successive generated training sessions. 0 if the trainer estimated that the exercises proposed in a session was not consistent at all with regards to the capacity and the sub-capacities trained, and 10 if the trainer was satisfied with the exercises proposed. The mean marks are reported in this figure. 6 trainers considered the sessions consistencies were between 7.2 to 7.8. Only the last trainer considered the mean consistency of the sessions was about 6.4. This was because AI-VT replaced many exercises by others, less important at the eyes of this trainer in the two last sessions.

The mean marks for the diversities of the training sessions are also reported in Figure 2. 6 of the trainers gave mean marks between 7.4 and 8 to this aspect. There was only one mark of 6.8 for one trainer. This was due to the second session generated for this trainer, in which most of the exercises were the same as the ones proposed in the first session. This was due to the initialisation of the RD of the exercises and sub-capacities. Indeed, if these RD are too high for some sub-capacities and exercises, AI-VT will propose them until other exercises have a higher RD .

4.2 Evaluation of AI-VT in the context of computer science learning (algorithmic)

For this second evaluation, we have proposed to 7 learners of computer science to use AI-VT for their training. These learners of our university (first year of study) are actually in difficulty with algorithmic, and they take tutoring sessions. We proposed them to resolve the exercises generated by AI-VT for 4 consecutive weeks, and 1 session of 60 minutes per week. After each training session, we asked them to evaluate the session generated by the system through the same aspects: consistency and diversity of the proposed exercises.

The mean marks obtained by AI-VT are reported on Figure 3. The mean mark obtained by AI-VT for consistency is 6.56. The consistency of the training session was not so good because the learners were very disappointed by each repetition of exercise. Some of the learners also found that the exercises were not adapted to their initial level (particularly in the first training session). Indeed, it would be appropriate to evaluate the levels of the learners before the first training sessions in order to propose exercises with appropriate difficulties since the first time a capacity is worked. This is the main reason why there are so many differences on AI-VT evaluation from a learner to another.

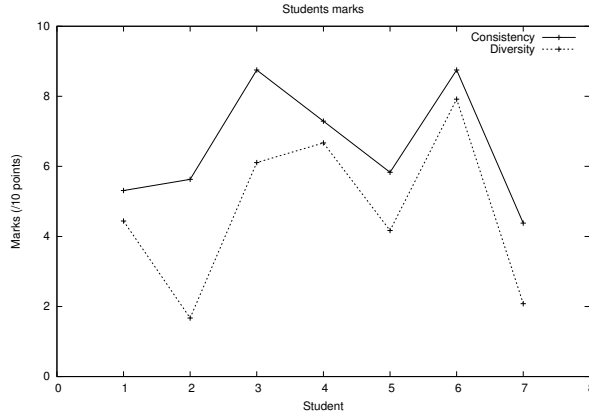


Fig. 3. Consistency and diversity marks obtained by the training sessions to algorithmic generated by AI-VT

The mean mark for the diversity is 4.72. Even if the system did not generate twice the same training session, the learners were disappointed when AI-VT proposed the same exercise twice or more during the cycle.

5 Discussion

AI-VT is a distributed CBR-based system that meet most of the requirements of its application domain. Indeed, in the particular field of sports training, the system can propose varied and consistent training sessions for many weeks even if the same capacity is requested several times consecutively. The consistency of the set of exercises proposed for each training session is guaranteed by the introduction of exercise complexity and distances between the different techniques. Hence, these distances allow the system to propose exercises that were not initially chosen by the trainers and to sort the exercises in each course by giving them an underlying theme. The diversity introduced by the system may sometimes influence badly the consistency of one session. Indeed, AI-VT may substitute an exercise by a less appropriate one, in order to satisfy the diversity requirement.

The performance of the system is less important for theoretical training. Indeed, in that particular field, even if the learners who tested AI-VT were globally satisfied, they were really disappointed when the system proposed the same exercise twice or more. Indeed, in that particular field, the diversity of the proposed training sessions was not satisfying. This is due to the capitalisation process which does not clearly dismiss the past resolved exercises. As a consequence, we must improve the formulas for the capitalisation phase so that exercises that were partially resolved or even resolved with high difficulty should not be proposed more than twice to the learners, and not proposed in consecutive training sessions. Nevertheless, in this theoretical field of algorithmic learning, all the

training sessions were consistent and all the proposed exercises were appropriate.

AI-VT establishes a link between the adaptation and capitalisation of CBR-systems. Capitalisation is of the utmost importance in this system which is required to give varied and creative solutions each time. Indeed, we have designed a way to use the remarks made by users during the revision phase in order to enhance the accuracy of the adaptation process of CBR-systems. In addition, the introduction of remaining durations is of the utmost importance since it allows AI-VT to build varied solutions by analogy and thus to never propose the entirely same session twice. In addition, the initialisation process of AI-VT is time-consuming for the teacher. Indeed, the teacher has to organise its sessions and exercises in capacities and sub-capacities and give the distances and the complexity of the exercises stored. In addition, the diversity of the exercises proposed in the training sessions depends on the number of stored exercises. For that reason, we will study the possibility to generate exercises automatically and a way to help the trainer to determine the initial values of the remaining durations (*RD*) of each exercise and sub-capacity.

6 Conclusion

We have designed a system based on Case-based reasoning and dedicated to the generation of varied training sessions for learners. AI-VT meets one of the most important requirement: its ability to generate varied training sessions. With this implementation of AI-VT, we highlighted the importance of the capitalisation phase of the CBR for the system diversity. Indeed, this capitalisation phase stores the story told to the learners, i.e. the training sessions and training exercises stored. The process that stores these training session has an impact on the fact that an exercise should be proposed once again or not. In addition, we proved the ability of AI-VT to adapt the exercises diversity of the training session generated to the context of use. Indeed, the capitalisation phase of AI-VT is adapted to the context and type of learning it is used for. The results obtained for sports training are very different from the ones obtained for theoretical learning like algorithmic. In the case of sports, the learning can be based on the repetition of the same exercises time after time. Indeed, even if an exercise is proposed at the beginning of each training session, it helps to automate certain actions. On the contrary, being confronted to the same algorithmic exercise twice or more is disappointed for learners since they already have the resolution of the exercise stored somewhere on their computer. As a consequence, even if the process of generation of a training session is based on analogy with past situations, an accurate tension between repetition and diversity is proposed by AI-VT, depending on the learned field.

Acknowledgement

The authors wish to thank the FR-EDUC (*Research Federation for Education*) of University of Franche-Comté for their financial help with this project.

References

1. Bernstein, N.: The co-ordination and regulation of movements. The co-ordination and regulation of movements (1966)
2. Biswas, G., Leelawong, K., Schwartz, D., Vye, N., at Vanderbilt, T.T.A.G.: Learning by teaching: A new agent paradigm for educational software. *Applied Artificial Intelligence* **19**(3-4), 363–392 (2005)
3. Gero, J.S., Maher, M.L.: Modeling creativity and knowledge-based creative design. Psychology Press (2013)
4. Gisolfi, A., Loia, V.: Designing complex systems within distributed architectures: an intelligent tutoring systems perspective. *Applied Artificial Intelligence* **8**(3), 393–411 (1994)
5. Grace, K., Maher, M.L.: Surprise-triggered reformulation of design goals. In: AAAI. pp. 3726–3732 (2016)
6. Grace, K., Maher, M.L., Wilson, D.C., Najjar, N.A.: Combining cbr and deep learning to generate surprising recipe designs. In: International Conference on Case-Based Reasoning. pp. 154–169. Springer (2016)
7. Graesser, A.C., Conley, M.W., Olney, A.: Intelligent tutoring systems. *The APA Educational Psychology Handbook* **3**, 451–473 (2012)
8. Huang, M.J., Huang, H.S., Chen, M.Y.: Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach. *Expert Systems with Applications* **33**(3), 551–564 (2007)
9. Jamsandekar, P., Patil, M.: Online learning - cbr approach. *International Journal of Research in Computer Science and Information Technology* **1**, 111–113 (2013)
10. Kolodner, J.: Case-based reasoning. CA Morgan Kaufmann (1993)
11. Kolodner, J.L., Cox, M.T., González-Calero, P.A.: Case-based reasoning-inspired approaches to education. *The Knowledge Engineering Review* **20**(03), 299–303 (2005)
12. Kolodner, J.L., Owensby, J.N., Guzdial, M.: Case-based learning aids. *Handbook of research on educational communications and technology* **2**, 829–861 (2004)
13. Maher, M.L., Grace, K.: Encouraging curiosity in case-based reasoning and recommender systems. In: International Conference on Case-Based Reasoning. pp. 3–15. Springer (2017)
14. Müller, G., Bergmann, R.: Workflow streams: a means for compositional adaptation in process-oriented cbr. In: International Conference on Case-Based Reasoning. pp. 315–329. Springer (2014)
15. Rishi, O., Govil, R., Sinha, M.: Distributed case based reasoning for intelligent tutoring system: an agent based student modeling paradigm. *World Academy of Science, Engineering and Technology* **5**, 273–276 (2007)
16. Schank, R.C., Fano, A., Bell, B., Jona, M.: The design of goal-based scenarios. *The Journal of the Learning Sciences* **3**(4), 305–345 (1994)
17. Tan, X.h., Shen, R.m., Wang, Y.: Personalized course generation and evolution based on genetic algorithms. *Journal of Zhejiang University SCIENCE C* **13**(12), 909–917 (2012)
18. Tyler, R.W.: Statistical methods for utilizing personal judgments to evaluate activities for teacher-training curricula... Ph.D. thesis, University of Chicago (1927)