AI-VT: an educative system based on artificial intelligence and designed to produce varied and consistent training lessons

Julien Henriet, Christophe Lang, Laurent Philippe
FEMTO-ST DISC - Equipe CARTOON

August 25, 2016

abstract

AI-Virtual Trainer (AI-VT) is an educative system based on Artificial Intelligence (AI) that proposes varied lessons to trainers. The agents of this multi-agent system apply case-based reasoning to build solutions by analogy. However, as required by the field, AI-VT never proposes the same lesson twice, whereas the same objective may be posed many times consecutively. The adaptation process of AI-VT delivers an ordered set of exercises adapted to the objectives and sub-objectives chosen by trainers. This process has been enriched by including the notion of distance between exercises: the proposed tasks are not only appropriate but are hierarchically ordered. With this new version of the system, students are guided towards their objectives via an underlying theme. Finally, the agents responsible for the different parts of lessons collaborate with each other according to a dedicated protocol and decision-making policy since no exercise must appear more than once in the same lesson. The results prove that AI-VT, however perfectible, meets the requirements of this field.

1 Keywords

Educative system, sports training, artificial intelligence, distributed case-based reasoning, multi-agent system.

2 Introduction

Motivation is a key aspect in teaching. Repetitivity causes students to be bored and to "turn themselves off" with respect to lessons. Nevertheless, repetitivity is of the utmost importance in acquiring skills. In addition, teaching cannot be reduced to a set of exercises. The exercises must be sorted according to complexity and each exercise must prepare students for the next one. At the same

time, it is necessary to treat all the aspects of a field, to introduce originality and diversity, and to adapt the exercise level and nature to the students' already acquired skills. This is particularly true in sports, where trainers must propose varied exercises and consistent lessons while having to train for the same skill over a given number of weeks.

This paper presents AI-Virtual Trainer (AI-VT), our education system based on Artificial Intelligence (AI). AI-VT is a Multi-Agent System (MAS) that uses Case-Based Reasoning (CBR) to provide consistent lessons with widely differing progressions. CBR is a problem solving method that adapts solutions from similar known problems in order to solve new problems [1]. CBR is widely employed in e-learning systems and Intelligent Tutoring Systems (ITS) [2]. J. L. Kolodner [3] distinguished between two types of CBR-inspired approaches to education: Goal-Based Scenarios [4] where learners achieve missions in simulated worlds thus confronting themselves with the real world, and Learning By Design [5] in which learners design and build working devices to obtain feedback. CBR is actually well-suited to the latter type of system [6], as well as to other tools from AI [7] and Distributed AI (DAI) systems [8] such as Genetic Algorithm (GA) [9], Artificial Neural Network (ANN) [10] and MAS [11]. A. Baylari and G. A. Montazer focused on the adaptation of tests to obtain a personalised estimation of a student's level [10]. They used an ANN in order to correlate student 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 [12]. O. P. Rishi et al. designed an ITS based on agents and a CBR system [13] in which a Personal Agent is responsible for determining student level. A Teaching Agent then determines the educative strategy with the help of CBR regarding the description of the transmitted student 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 students 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.

Many studies have highlighted the advantages of applying these AI concepts to sports training and physical activities [14], [15], [16], [17]. According to their classifications, AI-VT is an expert and knowledge-based system that aims at generating plans for training sessions according to agreed training strategies. Like MOPET [18], MAS are often used in order to collect and analyse real-time data capted by devices worn by athletes. Mathematical models analyse movements and help athletes to optimise their techniques [19]. Ambient intelligence systems have also been used in order to adapt training sessions [20]: J. Vales-Alonso *et al.* present a prototype for cross country running scenarios that maintains the heart rates of runners within target ranges. This system's decision tool is based on spline interpolations that take environmental parame-

ters (temperature, track hardness) and the athlete's heart rates as parameters in order to select the next track in the running scenario. Mathematical models based on past experiences (polynomial interpolators) have been developed by I. Rygula and J. Cholewa in order to optimise the parameters of training loads [21]. Maszczyk *et al.* explored the possibility to use ANN in order to predict performance of javelin throwers [22]. A. Erdogan *et al.* also used ANN interpolations in order to predict the heart rate at the onset of blood lactate accumulation due to external parameters [23]. Actually, each interpolation tool of these systems (ANN, spline and polynomial) optimises one particular aspect of the athlete's training, whereas AI-VT is designed to deal with many objectives and many abilities simultaneously.

The particular fields of sports training and physical education must take into account the component exercises in lessons in order to help maintain motivation among athletes/students. Many exercises are stored in the AI-VT knowledge base. Each exercise may be linked to one or many sub-objectives (or to none at all), and each sub-objective is linked to one or many objectives. A lesson has a single objective divided into sub-objectives, with each sub-objective divided into exercises. Each agent in this system is in charge of proposing exercises for one sub-objective and must take into consideration the exercises proposed by the other agents. This introduces another difficulty since an exercise must not be proposed more than once in the same lesson. This new version of AI-VT introduces the notion of distance and order of exercises in the resulting lesson. As highlighted by the results of our previous investigations [24], each new exercise must be as close as possible to the preceding and following ones. An underlying theme must be evident and hierarchical difficulty must be taken into consideration in the sorted set of exercises. Thus, solutions proposed by all the agents must be chosen collectively, taking into account the training history (the previous lessons proposed to the athlete during the season) as well as the solutions proposed by all the other agents. Therefore, AI-VT agents are collaborative entities.

In the next section, we present the purpose, the requirements, the knowledge structures and the methods involved our distributed system AI-VT. System implementation and performance are then presented and analysed.

3   Materials and Methods

In the first part of this section, we detail the requirements and the lesson structure of our distributed system. The distributed architecture and the data flows are examined in the second part. Finally, in the third part, we present how a lesson is designed theoretically.

3.1   Lesson structure

We considered physical activities for which training lessons are guided by the objectives [25]. Trainers have divided the courses into cycles of two to seven

weeks, with a minimum of two lessons per week. One specific skill is chosen for each cycle, and the same objective can consequently be assigned to a tenth of consecutive lessons. The chosen objective is then divided into elementary abilities that have to be mastered by the student/athlete and the mastery of each skill becomes a single sub-objective. Indeed, each skill may be shared by more than one objective and the mastery of each skill is a time consuming process that is reached through the repetition of exercises. The delay required for mastering a single skill depends on the student. Some will learn faster than others and thus the trainer must adapt each lesson to the level of the group. Consequently, a lesson is a sorted set of exercises carefully chosen by the teacher. Some exercises are more complicated or fundamental than others. A lesson usually begins with simple exercises that prepare the student for the next ones. Each exercise must be accessible on the basis of the preceding one and lead the student to the following one. This implies that a certain complexity must be associated with each exercise and that calculated distances between exercises must be established.

Consequently, AI-VT must:

(i) propose pertinent sub-objectives and exercises in function of the objective decided by the trainer and the level the students have reached via their previous lessons,

(ii) ensure that no exercise is proposed more than once during a given lesson and that the lessons in the same training cycle (to reach a single objective) are varied,

(iii) build a consistent course such that it begins with the simplest exercise to then continue with a list of exercises sufficiently related to the preceding and following ones, which implies that the system is capable of replacing each exercise initially proposed by another one that fits better without any intervention from the trainer, and

(iv) help the trainer to determine the importance of each sub-objective and exercise and the time to be devoted to them.

3.2   System architecture and communication model

MAS constitute a paradigm designed to handle distributed systems. They are the product of AI research and reflect its limits: a single AI representing the behaviour of a unique entity cannot deal with collective behaviour. Thus the idea of intelligence distribution appears but cannot be called DAI. In a 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.

As explained in the previous section, the choice of sub-objectives regarding an objective takes place via an autonomous process, as does the determination

of exercises regarding a sub-objective, or of any other exercises chosen and their priority level. The initial choice of exercises regarding a sub-objective must be an autonomous process in order to ensure a wise and free selection of the most adapted exercises. These processes can be undertaken simultaneously, coming after the determination of sub-objectives. Each process is based on specific rules and reasoning. In addition, each one must interact with the other processes and take their choices into account. Thus, each process must be autonomous, make decisions, infer changes from the choices made by the others, be aware of its environment, communicate and interact with the others. Consequently, we can call them agents.
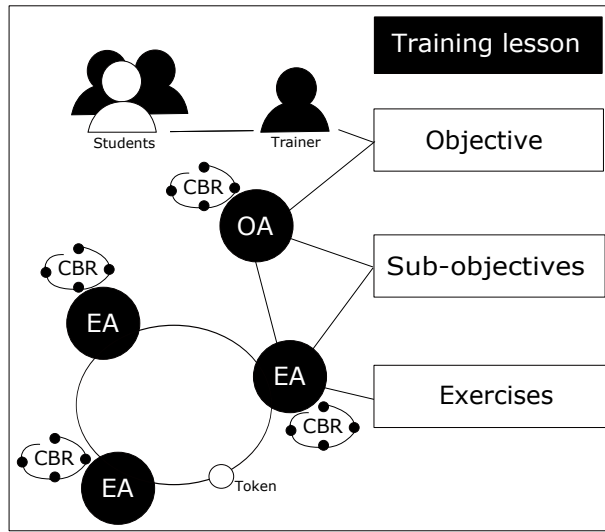

Figure 1: Communication model of AI-Virtual Trainer.

As shown in Figure 1, the system is composed of four types of agent: the trainer agent, the student agents, the Objective Agent (OA) which is responsible for choosing the sub-objectives regarding an objective requested by the trainer, and the exercise agents. Each of these agents is responsible for proposing the exercises the best suited to a given sub-objective. A single sub-objective is proposed by the OA or by another Exercise Agent (EA). Each EA must also take into account the choices made by the other EAs: each exercise may be done only once during the entire lesson. Thus, the choices of the EAs are shared and a decision policy is designed and tested.

To share the memory among the EAs, we have chosen the Pilgrim protocol which is an efficient and secured protocol for concurrent, cooperative and collaborative works with shared memory [26]. The agents are dispatched over a ring and a token is exchanged. The originality of this protocol is the fact that the token carries each stored modification. Each agent has a copy of the shared memory (i.e. the set of exercises proposed by the other agents). In

this protocol, the token is a structured entity that is transmitted from agent to agent and dispatched over a logic ring in an order determined at the beginning of their cooperation. When an agent wants to modify the shared memory, it can indicate its intention by placing a reservation flag above the token as soon as it receives it, on condition that the token has no other flag above it. The token continues its course over the ring and when the agent again receives the token, it is allowed to rewrite the modifications. Thus, all the other agents receive these modifications during the token's next revolution, after which all the agents have exactly the same version of the shared memory, even if two or more agents wanted to modify it at the same time. The decision policy is implemented over each EA that is able to modify its set of proposed exercises, if one or more of its exercises are identical to any of those found in another set.

Consequently, AI-VT is composed of two types of CBRs: CBR that proposes sub-objectives regarding objectives, and CBRs that propose exercises regarding sub-objectives. The next two subsections present these two types of CBRs.

3.3 Determination of sub-objectives

The OA is responsible for choosing the set of sub-objectives and their duration. Once the lesson has been chosen by the trainer, and after having analysed any additional student needs, the OA follows the CBR approach to make these choices in function of the sub-objectives already achieved and according to the student's degree of assimilation.

For the OA, a case is a set comprised of two parts: a problem and a solution. Each problem part is composed of an objective O, and the solution part of a set of $(SO, D_{SO,O})$ where SO is a sub-objective and $D_{SO,O}$ the duration required to reach this SO regarding O. Thus, formally, a source case s is expressed as $s = (O, \bigcup\{SO, D_{SO,0}^S\})$ The durations $D_{SO,O}$ are initialised by the trainer at the beginning of the season and updated by AI-VT after the lesson taking into account the remarks made by the trainer. Since student levels of expertise rise, we can consider that durations decrease and thus call them *remaining durations*.

First, all the sub-objectives associated to O are retrieved. The adaptation phase consists of computing the duration of each sub-objective. We assumed that these durations are somehow linked to the importance of practising each sub-objective regarding the given objective. Consequently, the adaptation module sorts the set of sub-objectives according to $D_{SO,O}$ (descending order). Then the proposed durations are calculated according to the number of sub-objectives the trainer wants to work on.

Following the lesson, the trainer evaluates the students' acquired level of mastery. Before the lesson begins, each selected sub-objective is transmitted to one EA that will have to associate the corresponding exercises. After the lesson, each sub-objective duration is modified in proportion to the evaluation from 0 to 10 of the students' level for the proposed sub-objective.

Finally, during the capitalisation phase, the system subtracts the durations from all the durations of the source cases for which the practiced sub-objectives

Table 1: Example of modifications of durations after a lesson.

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

appear.

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-objectives. Since the trainer chose the objective *become capable of 'Using a grip'* for this lesson, the OA recalled case 1. The adaptation process then sorted the sub-objectives according to their durations and proposed the three first sub-objectives, allocating $D_{SO} = \frac{60}{3} = 20$ minutes to each sub-objective. Consequently, after capitalisation, the new durations were those reported in the last column of Table 1. Thus, the less assimilated sub-objectives (*'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 objective (*'Using a grip'*) was selected again, another set of sub-objectives (composed of the less assimilated ones and others) were selected. Thus, as required, the proposed solutions changed even if the same objective was requested again later.

3.4 Selection of exercises

This subsection presents how the exercises are chosen regarding the selected sub-objectives. The different steps of the exercise selection process are reported in Figure 2.

3.4.1 Case model

For this part of AI-VT, the problem part of a case is composed of a sub-objective while the solution part contains a set of exercises with a specified duration to be spent on them. Though, exercise durations are constant, as for sub-objectives the total remaining duration required for each exercise must be specified and will decrease from one lesson to the next. Consequently, as presented above, each source case is noted:
$\sigma = (SO, \bigcup \{(EX, CD_{EX}^{\sigma}, RD_{EX,SO}^{\sigma})\})$.
Each source case σ contains the exercises possible regarding SO. 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 lesson that will be proposed) taken into account by the
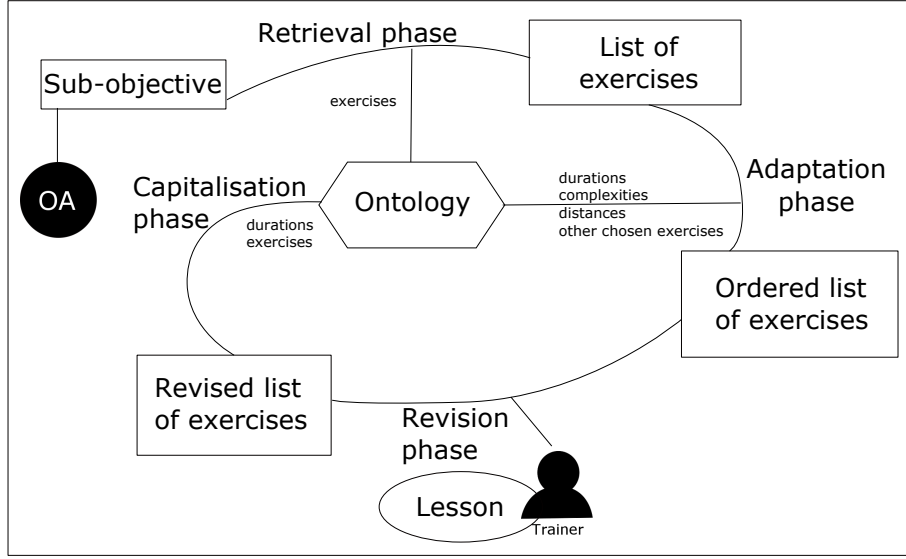
Figure 2: Process of exercise selection.

EA EA_i is noted:
$$\tau_i = (SO_i, \bigcup_{n \in \{1..Card(Sol_\sigma)\}} \{(EX_n, CD^{\tau_i}_{EX_n}, RD^{\tau_i}_{EX_n, SO_i})\}).$$

3.4.2 Retrieval phase

EA_1 is the EA initiated by the trainer. Hence, its role also consists of initiating as many EAs as required, and creating the token of the Pilgrim protocol. Each EA_i then retrieves the source case corresponding to SO_i.

3.4.3 Adaptation phase

The adaptation phase is the most important. Indeed, as highlighted by the analysis of our previous version of the system, the retrieval of accurate exercises is not sufficient. AI-VT must propose a consistent and a progressive list of exercises. In order tosupply this drawback of our previous version of AI-VT, we have introduced now the notion of distance between exercises. Consequently, the new elaboration of a course now follows five stages.

The first stage is based on the remaining durations of the techniques stored in the ontology. We assume that the exercises with the highest remaining durations are the most important to practice, or must be practiced regularly to reach a sub-objective. For this reason, the exercises are decreasingly ordered according to their remaining duration and each one is added until the sum of constant durations reaches the practise time required to attain sub-objective SO_i, i.e. $(\sum_n CD^{\tau_i}_{EX_n, SO_i}) \geq D^{T,i}_{SO_i, O}$. Before placing its list of exercises on the token, EA_i verifies whether each exercise has been selected with a higher or equal

duration by another EA. If so, EA_i removes its exercise and replaces it with the next unselected one. EA_i may not have enough remaining exercises for the duration period and thus places a $LackOfExercise$ instead.

The second stage is based on the distances between the exercises. Indeed, the trainers are invited to initialise the distances between each exercise and every other one. At the end of this stage, AI-VT replaced each $LackOfExercise$ by the unselected exercise closest to the one initially selected (proposed by another EA).

The third stage is based on the difficulty of the exercises also set by the trainer during the initialisation process of AI-VT. As a result of this stage, the system proposes an ordered list of exercises of increasing complexity. Within each sub-objective, the associated exercises are also ordered by increasing complexity.

During the fourth stage, the distances are considered once again. Each EA must verify that the distance between each exercise it has proposed and the previous one is lower than a threshold set by the trainer. If this is not the case, the exercise must be exchanged with the closest unused exercise that respects this distance constraint. During this process of choice, the distance between the new exercise and the previous one in the lesson, and the distance between this new exercise and the next one in the lesson must both be lower than the required threshold. This process may take a long time since all the EAs manipulate their exercises simultaneously.

Finally, during the fifth stage, each EA tries to switch the places of their consecutive exercises in order to optimise the distances between the consecutive exercises. During this stage, it is strictly forbidden to propose other exercises. Only their order can be changed. Hence, an exercise cannot be associated to another sub-objective. If another order that minimises the distances between two consecutive exercises is found, it is retained.

The adaptation phase requires the EAs to communicate with each other. They must associate ordered lists of exercises with sub-objectives sharing their solutions according to the Pilgrim protocol which has been enriched so that each stage ends with the acknowledgement of all the EAs. Each EA must wait for the acknowledgement of the others before going on to the next stage of the phase.

3.4.4 Revision phase

Before and after the lesson, the trainer can create, modify, replace, change the order of or remove exercises and durations. At the end of the course, the trainer evaluates the students' level. The same computations as those applied to the sub-objectives regarding the objective are applied to the remaining exercise durations:
$\forall i, \forall n \in \{1..N_i^{EX}\}, CD_{EX_n,SO_i}^{\tau_i} = CD_{EX_n}^{\tau_i} \times \frac{level_{EX_n}}{10}$.

3.4.5 Capitalisation phase

All the new exercises are added to the corresponding case and the new remaining durations are stored in the sub-objective/exercise ontology:
$$\forall SO, \forall EX, RD^{\sigma}_{EX,SO} = max(0, (RD^{\sigma}_{EX,SO} - CD^{\tau}_{EX,SO})).$$

4 Results

As a means to evaluate AI-VT, four aikido trainers initialised the system for the same objective and requested ten consecutive lessons. At the end of each lesson, they evaluated the solution proposed by AI-VT and gave a mark (out of 20 points) regarding four aspects (each one out 5 points):

(i) the consistency of the techniques (exercises) proposed regarding the sub-objectives,

(ii) the variety of techniques,

(iii) the progressivity of the techniques proposed, and

(iv) the connections established between the different techniques (prior and/or following).

A score of five points means the lesson meets its goals perfectly. Zero means that it was not satisfactory at all. The marks are reported in Figure 3. Actually, all the lessons proposed were very satisfying for all the trainers since all the marks vary from 12 to 16 points, with the mean mark totalling 14.65 and the median mark equal to 15. The lowest marks were given to lessons for which the techniques substituted by AI-VT were not a good match to the given sub-objective. For example, one of the sub-objectives of Trainer #2 was 'Using a back side grip'; sometimes AI-VT proposed techniques that were close but front side grabs. This implies that a semantic element must be introduced in order to ensure that one exercise can be substituted for another and still correspond to the associated sub-objective.

Table 2 shows the sequence of the lessons that AI-VT proposed to one of the trainers (Trainer #1). He initialised the system with four sub-objectives and ten techniques associated with each of the different sub-objectives. Some of the techniques were associated with two sub-objectives. This trainer asked for ten lessons, each containing three sub-objectives, and for each sub-objective to be practiced through two aikido techniques.

The orders of appearance of each sub-objective are reported in the first part of Table 2. The sub-objective that appears with the lowest frequency is actually the one that was associated with the simplest techniques ('Making the partner lose balance' appears 6 times). Thus, this technique was always proposed first among the lessons and the trainer gave the students their highest marks. Consequently, this sub-objective quickly became less important than the others during AI-VT's retrieval phase. The sub-objective 'Relaxing despite
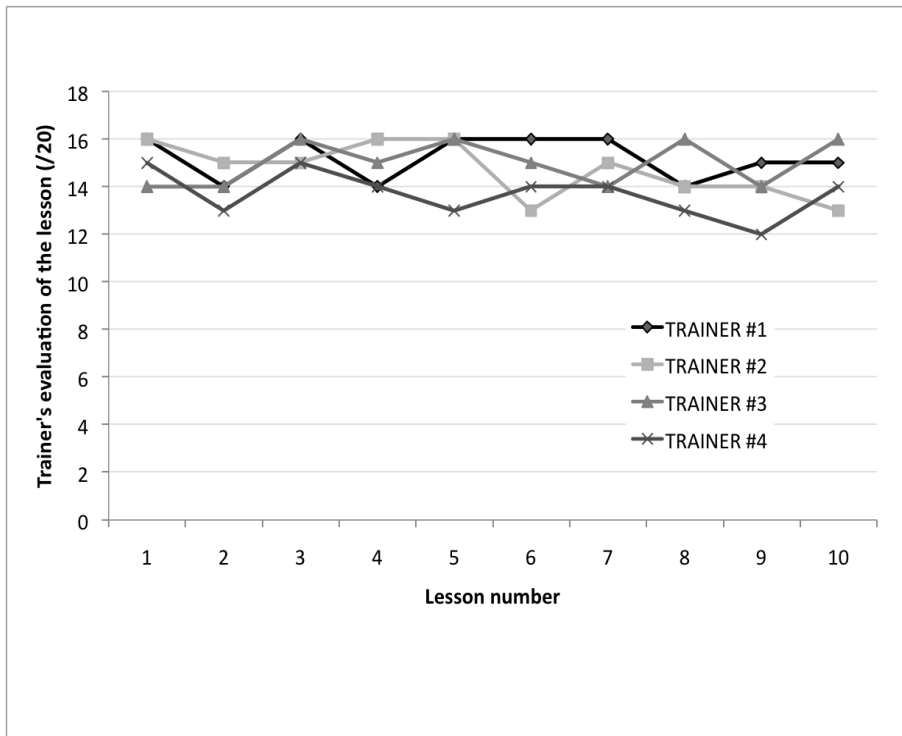
Figure 3: Trainer's evaluations of the generated lessons.

Table 2: Lessons proposed by AI-VT to Trainer #1.

Skill	Order of appearance in Lesson									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Making the partner lose balance	1	1		1		1		1		1
Breaking the partner's posture	2	2	1		1	2	1		1	
Relaxing despite a grip	3	3	3	3	3		3	3	3	3
Pivoting around a grip			2	2	2	3	2	2	2	2
Aikido technique										
KTD IO	1			2		2		2		2
AHKD IU	2	2				1		1		
AHKD IO	3	1		1						1
KRD KNIO (*Syst.*)	4					3				
KRD UKN	5	6	6				5		5	
KRD AO	6						6		6	
KRD IN (*Syst.*)		3								
KTD TN		4	2		1		1		1	
KRD KH		5	5			4				
KTD SIN			1		2		2		2	
KTD UKNO			3				4			6
RD KNS (*Syst.*)			4							5
RD IN (*Syst.*)				3	3		3	3	3	3
URD SU				4	4	6		4	4	4
AHKD KNIO (*Syst.*)				5	5			5		
KRD KSNIO (*Syst.*)				6						
KRD IO (*Syst.*)					6					
AHKD IN (*Syst.*)						5				
KRD KSNYO (*Syst.*)								6		

a grip' is the most frequent one (AI-VT proposed it 9 times out of 10) owing to the low student level: according to the trainer, this technique is actually one of the most difficult skills to master. The other two sub-objectives appeared 7 and 8 times; since only four sub-objectives were to be distributed over ten lessons, each required three sub-objectives.

Nevertheless, the lessons were not repetitive since AI-VT never proposed the same ordered set of exercises (nor the same set of unsorted exercises) twice. The techniques proposed by the system are shown in the second part of Table 2: the numbers correspond to the order of appearance of each technique in the corresponding lesson. The names of the aikido techniques are not fully reported, only their acronyms composed of the first letters of the words. In order to quantify the diversity of the solutions proposed by AI-VT, we calculated the Shannon index [27], a value widely used in ecology in order to quantify species diversity. Considering that AI-VT delivered 10 lessons, each having 6 aikido techniques selected out of 19 possible techniques, and assuming N_i is the number of times of appearance of technique i, the Shannon index H is computed according to the equation: $H = -\sum_{i=1}^{10} \frac{N_i}{19} \times ln\frac{N_i}{19}$. The Shannon entropy is therefore equal to $H = 6.77$ and quantifies the uncertainty associated with the prediction of retrieving each technique in one of the generated lessons. In order to quantify the lesson diversity, it is necessary to divide H by $H_{max} = -ln(p)$ where p is the probability that each technique would have been chosen for each lesson, assuming that each technique would have been chosen in each lesson with the same probability (which is an erroneous assumption considering the way AI-VT chooses techniques): $H_{max} = -ln(\frac{1}{\mathcal{A}_{19}^6}) = ln(\mathcal{A}_{19}^6)$ where $\mathcal{A}_{19}^6 = \frac{19!}{(19-6)!}$ is the number of arrangements of 6 out of 19 techniques. Consequently, since $\frac{H}{H_{max}} = \frac{6.77}{ln(\mathcal{A}_{19}^6)} = 0.4033$ is a quantity closer to 0 than 1, it proves the high level of diversity of the sets of techniques of this sequence of lessons.

In the first column of Table 2, we added '(*Syst.*)' after the acronym of the techniques that the trainer did not associate with any of the sub-objectives: these techniques were added by AI-VT itself during the adaptation phase. This table shows that all the techniques initially associated with the sub-objectives were proposed at least three times, and AI-VT proposed 9 new techniques from the aikido catalogue that were not initially associated with any of the trainer's defined sub-objectives. The trainer found that these new techniques were always pertinent choices for reaching the sub-objectives and compatible with the neighbouring techniques. It is noteworthy that the most frequently proposed techniques (6 times) are initially proposed by the trainer ('*URD SU*') and also by the system, initially ('*RD IN*'), but not by the trainer. This is due to the fact that these two techniques are very close to much of the initial set of techniques proposed by the trainer, and thus can easily be added to every lesson with no impact on the lesson consistency. The other techniques that are proposed 4 times or more ('*KTD IO*', '*AHKD IU*', '*AHKD IO*', '*KRD UKN*', '*KTD TN*', '*KTD SIN*') are considered by the trainer to be fundamental exercises, and the fact that they were often proposed in the AI-VT lessons reassured him. In contrast, the techniques that appeared only once or twice were not initially chosen by

the trainer; AI-VT substituted other techniques for them. These choices were nevertheless pertinent according to the trainer and regarding the sub-objectives to be reached.

5   Discussion

AI-VT is a distributed system that meets the requirements of the application domain. Though tested in the context of aikido training, it could easily be adapted to other sports. The system can propose varied and consistent training lessons for many weeks even if the same objective is requested several times consecutively. The consistency of the set of exercises proposed for each lesson 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. Like other approaches [28], [29] and [30], ours establishes a link between the adaptation and capitalisation of revisions of CBR-systems. 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 same lesson twice. Another originality of AI-VT resides in the fact that its propositions are groupings of solutions from CBR-based agents. These agents must collaborate closely, share memory and take into account the solutions proposed by the other agents.

However, many facets of AI-VT may be improved. Firstly, regarding the sub-objectives, the distances between the exercises are not always sufficiently well considered to guarantee the consistency of the proposed exercises. Even though the exercises proposed by AI-VT were close to the exercises initially proposed by the trainer, they were not always consistent with respect to certain very specific sub-objectives. Consequently, we must introduce some semantics into the system so that it can substitute exercises that correspond to the meaning of the associated sub-objective. In addition, with each exercise AI-VT should also provide consistent instructions which depend both on the exercise and the sub-objective to be reached. Thus, when the system substitutes an exercise initially chosen by the trainer for one or its own choosing the trainer's first set of instructions is no longer consistent. This particular feature is linked to the previous one since it must be based on the sub-objective's meaning and the exercise's characteristics.

Finally, one limit to our system lies in the fact that it must be initialized every year at the beginning of the season or whenever no remaining duration is left. At that time, the trainer must produce all of the initial values. Further investigation will likely lead us to propose a process based on experience in order to compute initial values. Like other approaches, we will evaluate the possibility of basing these computations on pre-tests that evaluate student levels [12] and [9].

6 Conclusion

We have designed a distributed system based on artificial intelligence and that is capable of delivering varied lessons to trainers. AI-VT meets the requirements of the application domain and can easily be adapted to many areas in the field of sports education. Indeed, we have proved that AI-VT can propose ten consecutive lessons aimed at the same objective, each one different from the others even if the process of choice is based on analogy with capitalised courses and experience. The agents of AI-VT collaborate in order to provide the most suitable exercises for each part of each lesson. A collaboration protocol based on a token ring has been adapted and used, as well as a policy capable of merging and arbitrating between concurrent solutions proposed by the collaborative agents during the adaptation process. In addition, we have introduced a distance between the exercises and associated a complexity to each exercise in order to guarantee the progressivity and the existence of an underlying theme common to all of the exercises on the ordered list of exercises composing the lesson. This study proves the consistency of the collegial solutions proposed through its evaluation by four trainers.

The system could nevertheless be improved if it could take into consideration the exact meanings of the sub-objectives to be reached during each proposed lesson. Consequently, future investigation will focus on the system's ability to perceive and interpret sub-objective meanings in order to provide students with wisely devised instructions and exercises, even when the system replaces an exercise by the closest one during its adaptation phase. Finally, we will examine ways in which to introduce processes such as the pre-testing of students in order to reduce the system initialisation process that must be performed by the trainer.

References

[1] Kolodner J. *Case-based reasoning.* CA Morgan Kaufmann; 1993.

[2] Graesser A, Conley M, Olney A. *Intelligent tutoring systems.* Am Psych Ass, The APA Educ Psy Handbook. 1992;3:451-473.

[3] Kolodner J, Cox M, González-Calero P. *Case-based reasoning-inspired approaches to education.* Cambr Univ Press, Knowl Eng Rev. 2005;20(3):299-303.

[4] Schank R, Fano A, Bell B, Jona M. *The design of goal-based scenarios.* Taylor & Francis, J Learn Sci 1994;3(4):305-345.

[5] Kolodner J, Owensby J, Guzdial M. *Case-based learning aids.* Handb Res Educ Comm Techn. 2004;2:829-861.

[6] Jamsandekar PP, Patil MK. *Online Learning - CBR approach.* Intern J Res Comp Sci Inf Techn. 2013;1:111-113.

[7] Biswas G, Leelawong K, Schwartz D, Vye N. *Learning by teaching: A new agent paradigm for educational software.* Taylor & Francis, Appl Art Intell. 2005;19(3-4):363-392.

[8] Gisolfi A, Loia V. *Designing complex systems within distributed architectures: an intelligent tutoring systems perspective.* Taylor & Francis, App Art Intell. 1994;8(3):393-411.

[9] Tan X, Shen R, Wang Y. *Personalized course generation and evolution based on genetic algorithms.* Springer, J Zhejiang Univ SCI C 2012;13(12):909-917.

[10] Baylari A, Montazer G. *Design a personalized e-learning system based on item response theory and artificial neural network approach.* Elsevier, Exp Sys Appl 2009;36(4):8013-8021.

[11] Rishi OP, Govil R, Sinha M. *Agent based student modeling in distributed CBR based intelligent tutoring system.* Proc W Congr Eng Comp Sci. 2007;473-477.

[12] Huang M, Huang H, Chen M. *Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach.* Elsevier, Exp Sys Appl. 2007;33(3):551-564.

[13] Rishi OP, Govil R, Sinha M. *Distributed case based reasoning for intelligent tutoring system: an agent based student modeling paradigm.* W Acad Sci, Eng Techn. 2007;5:273-276.

[14] Bartlett R. *Artificial intelligence in sports biomechanics: new dawn or false hope?.* J Sports Sci Med. 2006;5:474-479.

[15] Owusu G. *AI and computer-based methods in performance evaluation of sporting feats: an overview.* Artif Intell Rev. 2007;27:57-70.

[16] Balague N, Torrents C, Hristovski R, Davids K, Araujo D. *Overview of complex systems in sport.* J Syst Sci Complex. 2013;26:4-13.

[17] Fister I Jr., Ljubic K, Suganthan P, Perc M, Fister I. *Computational intelligence in sports: Challenges and opportunities within a new research domain.* Appl Math Comp. 2015;262:178-186.

[18] Buttussi F, Chittaro L. *MOPET: A context-aware and user-adaptative wearable system for fitness training.* Art Intell Med. 2008;42:153-163.

[19] McNitt-Gray J, Sand K, Ramos C, Peterson T, Brown K. *Using technology and engineering to facilitate skill acquisition and improvements in performance.* Proc Inst Mech Eng, Part P: J Sports Eng Techn. 2015;229(2):103-115.

[20] Vales-Alonso J, Lopez-Matencio P, Gonzales-Castano F, Navarro-Hellin H, Banos-Guirao P, Perez-Martinez F, Martinez-Alavarez R, Gonzalez-Jimenez D, Gil-Castineira F, Duro-Fernandez R. *Ambient intelligence systems for personalized sport training.* Sensors. 2010;10:2359-2385.

[21] Rygula I, Cholewa J. *Application of a mathematical model in control of sport training.* J Med Inf Techn. 2004;7:105-113.

[22] Maszczyk A, Zajac A, Rygula I. *A neural network model approach to athlete selection.* Sports Eng. 2011;13:83-93.

[23] Erdogan A, Cetin C, Goksu H, Guner R, Baydar M L. *Non-invasive detection of the anaerobic threshold by a neural network model of the heart rate-work rate relationship.* Proc Inst Mech Eng, Part P: J Sports Eng Techn. 2009;223:109-115.

[24] Henriet J. *Collaborative CBR-based agents in the preparation of varied training lessons.* Int J Comp Sci Sport. 2014;13(2):2:4-19.

[25] Tyler R. *Statistical Methods for Utilizing Personal Judgements to Evaluate Activities for Teacher-Training Curricula.* 1995.

[26] Garcia E, Guyennet H, Henriet J, Lapayre JC. *Towards an optimistic management of concurrency: a probabilistic study of the pilgrim protocol.* Springer, Comp Supp Coop W Des II. 2006;51-60.

[27] Shannon C. *A mathematical theory of communication.* Wiley Online Libr, Bell Sys Tech J. 1948;27:379-423 and 623-656.

[28] Cordier A, Fuchs B, Mille A. *Engineering and learning of adaptation knowledge in case-based reasoning.* Springer, Man Knowl W Net. 2006;303-317.

[29] Dufour-Lussier V, Le Ber F, Lieber J, Nauer E. *Automatic case acquisition from texts for process-oriented case-based reasoning.* Elsevier, Inf Sys. 2013;40:153-167.

[30] Henriet J, Chatonnay P, Leni PE. *An iterative precision vector to optimise the CBR adaptation of EquiVox.* Elsevier, Eng Appl Art Intell. 2014;35:158-163.