ExPLoRAA: An Intelligent Tutoring System for Active Ageing in (Flexible) Time and Space*

Amedeo Cesta, Gabriella Cortellessa, Riccardo De Benedictis, and Francesca Fracasso

> CNR - Italian National Research Council, ISTC, Rome, Italy, {name.surname}@istc.cnr.it http://www.istc.cnr.it/it/group/pst

Abstract. The "Città Educante" project aims at radically rethinking the learning experience through advanced ICT technology to enrich and innovate didactic methods and tools. Among the project results, ExPLo-RAA is an Intelligent Tutoring System, specifically tailored for senior citizens, which, by integrating artificial intelligence and state-of-the-art ICT techniques, is able to support older adults during visits to cultural locations in a city. In particular, ExPLORAA integrates both the users' psycho-physiological aspects as well as geo-localization information and temporal constraints in the attempt to personalize the learning stimuli during the visit while favouring the concept of active ageing for the older people. After a generic introduction to the "Città Educante" project, this paper presents both ExPLORAA as a whole and some of the underlying solutions. Specifically, the paper shows some of the choices that have been made to solve problems related to temporal flexibility, supporting the dynamic adaptation of stimuli over time while ensuring the possibility for the users to further adapt a visit according to their current feelings. The paper describes both the choices made in the current system prototype and its embodiment in a concrete scenario which, by implementing a game similar to "treasure hunt", aims at fostering the physical and cognitive activity of the participating older people.

Keywords: Intelligent Tutoring System \cdot Active Ageing \cdot Temporal Flexibility.

1 Introduction

The "Città Educante" project¹ (the name means "city that educates" in Italian) aims at proposing new educational approaches, enriching and innovating methods and tools, overcoming the classical systems and the traditional role of educators. An Intelligent Tutoring System (ITS) [15, 18] is a computer system that, thanks to personalized stimuli to the learners, enables learning in an effective and meaningful manner. Such systems aim at radically rethinking the

^{*} Authors work is partially funded by MIUR under Cluster Program 2012. Send correspondence to *amedeo.cesta@istc.cnr.it*.

¹ http://www.cittaeducante.it

learning environment through the application of the most advanced ICT technology and represent, hence, a valid contribution to the project.

Common ITSs, typically, aim at replicating the benefits of one-to-one personalized tutoring in contexts where students would have access to one-to-many instruction from a single teacher (e.g., classroom lectures), or no teacher at all (e.g., on-line homework) [17]. Such systems are typically related to classical learning environment, hence neglecting the possibility to act in *time* (e.g., lifelong) and *space* (e.g., at school, in an outdoor environment, during leisure, etc.), thus overcoming the classical systems and the traditional "lessons".

Through the "Città Educante" project, the theme of *learning* is framed in relation to the response to social challenges linked to the renewal of the educational system, to be achieved by means of the implementation of new learning/teaching models and/or the optimization of the existing ones on the various areas of life and knowledge, as well as new systems/evaluation processes, in which the technology (platforms and web) becomes an enabling factor.

Specifically, the authors' goal in the project has been the one of thinking an incarnation of the "Città Educante" for the continuous education of older people. In particular, considering the more recent experiences in the interaction of elderlies with complex machines [6], keeping in mind previous experience in training elders [4] and crisis managers [2], we have built a new learning environment, called ExPLORAA (for ExPeriential LeaRning for Active Aging), aimed at improving the active aging and the participation in the social life of the elders living at home, in the community, and at work.

2 Televita: the Target Organization

While pursuing the objective of the project, which aims at reformulating the learning environments through the creation of platforms, services and ICT applications, we got in touch with several volunteering organizations addressing, specifically, elderlies' needs. Among the different organizations, in particular, Televita² is a volunteering association whose main objective is to maintain the elderlies active and motivated, leveraging upon individual aptitudes and/or competencies [5]. Most of the Televita's volunteers, in fact, are, themselves, elders who want to keep active by offering their abilities and competences to the organization. Although Televita's main activity consists in providing tele-assistance services (tele-friendship) and a 24h active helpline devoted to lonely elders who need support, it also manages several laboratories that involve elders both as attendees and "teachers". Examples include a computer lab, a tailoring lab, a cooking lab and an Italian language teaching for foreign people. Furthermore, the association organizes cultural events as concerts, museum visiting, theater, etc.

Among the offered services, we focused in particular on two specific activities that were both in line with the "Città Educante" concept and that are outlined in Figure 1:

² http://www.televita.org



Fig. 1. Examples of Televita activities.

- The AttivaMente laboratory: aims at keeping elders mentally active, so as to limit the cognitive decline associated with the advancement of age, by proposing them cognitive stimuli. Such stimuli, mostly consisting in general culture quiz and/or crosswords, are proposed to the elders in a context similar to a school lesson. Specifically, by relying on some previous knowledge of the involved persons, as well as on their interactions during the lesson, a teacher, a volunteer himself, yet with more experience than others, controls the course's progress and slightly adapt it to the specific context's needs. Since the stimuli are predetermined, however, such adaptations tend to be limited to the possibilities of the case. The use of artificial intelligence techniques, in this case, could support the personalized delivery of stimuli by taking into account previous knowledge of the participants as well as their interactions with the AI system increasing, compared to the classical case, the personalization capabilities.
- The Art History lessons and cultural events: analogously to the AttivaMente case, before attending to cultural events, some of the participants, according to their abilities and competencies, are asked to find information about some specific aspects of the event (e.g., a particular work of art within a museum, relevant historical happenings related to a visited site, etc.). Such information are then shared, in a "sort of" lesson with other participants in a classroom context and also outside during the event, enriching the overall knowledge of the group while encouraging the interaction among the members. Similarly to the AttivaMente case, such information may result to be limited and/or not customized to the specific members of the event. AI techniques, in this case, might offer the opportunity to enrich the cultural events experience by providing further personalized stimuli to the event participants, as well as interaction requests to test the level of engagement and actively stimulate them.

Combining the above activities represents an opportunity for the elders to keep themselves active while learning in time and space. Additionally, the use of AI can support the development of more effective and engaging learning experience.

3 The ExPLoRAA Intelligent Tutoring System

The idea of developing the ExPLORAA system is born as a consequence of a field experience. Specifically, by taking inspiration from a previous work [3], in which students were trained for managing crisis, the approach used within Ex-PLORAA is based on the idea of dynamically composing lessons through the use of automated planning [11]. In particular, starting from a static representation containing an high-level lesson track, initially stored in a database, the lesson is planned and dynamically adapted and personalized to the involved users. The idea of using the technology related to automated planning comes from the need to create a sufficiently extensive didactic experience to reproduce a large number of different situations which are, at the same time, characterized by a high variability of stimuli, aimed at increasing the involvement level of users. Automated planning, indeed, favors the generation of different lessons that would be too complicated to obtain with a simple pre-compilation of stories. The timelinebased approach to automated planning [14], in particular, represents the unifying element of the various modules by ensuring the dynamic adaptability of plans by promoting experiential learning.



Fig. 2. The ExPLORAA general idea.

From a high-level point of view, the main modules of the system are described in Figure 2. In particular, it is possible to distinguish between two kinds of involved **users**: the *students*, i.e., a group of people, potentially, of any age, interested in using the learning services offered by the ExPLORAA environment, and the *teachers*, i.e., users with special privileges who have the opportunity to observe students, monitor the progress of the lessons and of the overall learning environment. The above users interact with the ExPLORAA system which is composed of three functional blocks, intended as architectural subsystems, implementing the corresponding high-level functionalities: (i) the *user modeling*, whose goal is to create and maintain a user model and provide guidance for improving the learning process; (ii) the *lesson modeling*, whose role consists in combining the information from the previous subsystem and to create the customized lesson as well as to control its evolution; (iii) the *lesson presentation*, whose purpose is to represent the lesson through effective graphical interfaces. Additionally, it is worth highlighting that the proposed system provides users, whether students or teachers, the opportunity to adapt the learning environment's evolution in real time by interpreting their decisions. In fact, the architecture is based on a *sense-plan-act* paradigm implementing, in a continuous loop, the three primitives (a) *sense*, in which information is collected from questions and sensors, (b) *plan* where a lesson blueprint is created, using the information available, and (c) *act*, in which the action, chosen by the planning process, is actually executed.

It is worth highlighting that, by exploiting mobile technology, the lesson presentation module interacts remotely with the system allowing different learning modalities: (a) on-site learning, closer to the classical teaching, in which the technology is used as a support to the teaching in a classroom, with the aim to create richer lessons, and (b) *distributed learning*, in which the technology aims to support lessons outside the classroom during a practical experience. More specifically, in the on-site training modality, the system can be used by a group of students at the same time. This mode represents an extension to the classical learning method in which a teacher teaches to a group of students. In this case, however, compared to the classical approach, the teaching is enhanced by the introduction, within the lesson, of the ExPLORAA technology. Each lesson is instantiated by the teacher by defining specific *learning objectives* [12] that we call *qoals*. The system processes the lesson and presents, at proper time, the information to the students through the presentation tools. Students interact directly with the system, providing their answers to certain circumstances proposed by the system, and transmitting data from sensors available on the adopted devices (e.g., physiological parameters) enriching the users' models. Conversely, in the distributed training case, the lesson does not happen in a single physical room and is distributed among the students who are remotely connected to the system. The lesson is still instantiated by the teacher defining the specific learning objectives but may have variable, potentially infinite, duration. This kind of approach, compared to the previous one, is more innovative. Students interact directly with the system while on the move within the city, providing their answers to certain proactive stimuli proposed by the system as well as constantly transmitting data from the sensors available on the chosen devices (e.g., geographic location, physiological parameters, etc.). Sensor data, in particular, enriches the users' models which, in turn, adapt the lesson to the students resulting in a highly personalized learning experience.

4 Timeline-based planning to support dynamic lessons

Since most of the components of the ExPLORAA system strongly depend on temporal aspects, we have chosen to rely on a specific automated planning technique, called timeline-based, which allows to explicitly reason on time. Timelinebased planning, indeed, allows to reason about events in time and, hence, represents a valid tool for meeting our pedagogical needs. Planning a lesson, in particular, requires dispatching information at proper time. Additionally, reacting to users' interactions requires plan adaptation capabilities which can more hardly be achieved through other automatic planning techniques. Furthermore, the dynamic adaptation of the user profiles, which can take place on the different features that represent the user's model, can also be achieved through timelines.

As already said, timeline-based planning constitutes a technology that easily allows us to solve our problems. In order to better contextualize the choices that we have made and to better explain the different components of the system, however, it is worth introducing some basic formalism about constraint networks, on which timeline-based planning search strongly relies, and about the main concepts related to timeline-based planning. Specifically, the main ingredients of constraint networks are variables and constraints.

Definition 1. A variable is an object that has a name and is able to take different values.

A variable (whose name is) x must be given a value from a set that is called the *domain* of x and is denoted by dom(x). The domain of a variable x may evolve in time but is always included in a set called *initial domain*. Depending on the nature of these domains, variables can be distinguished between *continuous*, having an infinite initial domain usually defined in terms of real intervals, and *discrete*, whose initial domain contains a finite number of values.

Definition 2. A constraint is a restriction on combinations of values that can be taken simultaneously by a set of variables.

A constraint c is defined over a set of variables which constitute the *scope* of c and are denoted by scp(c). Finally, a structure composed of variables and constraints is called a *constraint network*.

Definition 3. A constraint network \mathcal{N} is composed of a finite set of variables, denoted by vars (\mathcal{N}) , and a finite set of constraints, denoted by cons (\mathcal{N}) , such that $\forall c \in cons (\mathcal{N}), scp (c) \subseteq vars (\mathcal{N})$.

Since constraint networks are fundamentals for timeline-based planning, it is worth introducing some further concepts without going into too much formal details. Specifically, an assignment of values to some or all the variables is called an *evaluation*. Furthermore, an evaluation is said to be *consistent* if it does not violate any constraint. An evaluation is said to be *complete* if it includes all the variables. Finally, given a constraint network, the problem of finding a consistent and complete evaluation is called Constraint Satisfaction Problem (CSP) (refer to [9, 13] for a comprehensive introduction to CSPs).

As regards timeline-based planning, the main data structure is the *timeline* which, in generic terms, is a function of time over a finite domain. Values on the timelines are extracted from a set of temporally scoped predicates (i.e., predicates endowed with extra arguments belonging to the Time domain \mathbb{T} , either real or discrete), with their parameters, called *tokens*. Formally,

Definition 4. A token is an expression of the form:

$$n(x_0,\ldots,x_k) @ [s,e,\tau]$$

where n is a predicate name, x_0, \ldots, x_k are the predicate's parameters (i.e., constants, numeric variables or object variables), s and e are temporal parameters belonging to \mathbb{T} such that $s \leq e$ and τ is a parameter (i.e., a constant or an object variable) representing the timeline on which the token apply.

The overall idea pursued in ExPLORAA consists in using such tokens for representing the planned stimuli. Compared to the general formalization above, however, we can afford some simplifications. Specifically, since the stimuli have no duration, the s and e variables of each token would always be equal. We address this by removing one of the two variables, e.g., the e variable. Additionally, since all the tokens will apply on a single "lesson" timeline, the τ variable would be a constant. For example, an expression st_0 () @[10:00] would represent a stimulus st_0 which is planned to happen at time 10:00.

It is worth noticing that the tokens' parameters, including the temporal ones, are constituted, in general, by the variables of the constraint network as introduced in the Definition 1. In order to reduce the allowed values for such parameters, and thus decreasing the system's allowed behaviors, it is possible, indeed, to impose constraints, as introduced in the Definition 2, among them (and/or between the parameters and other possible variables). Such constraints include temporal constraints, usually expressed by means of interval relations [1], binding constraints between object variables as well as linear constraints among numerical, including the temporal one, variables.

The set of tokens and constraints is used to describe the main data structure that will be used to represent the nodes of the timeline-based search space: the *token network*.

Definition 5. A token network is a tuple $\pi = (\mathcal{T}, \mathcal{C})$, where:

- $\mathcal{T} = \{t_0, \ldots, t_k\}$ is a set of tokens.
- C is a set of constraints, required to be consistent, on the variables of the tokens in T.

Additionally, tokens can be partitioned into two groups: *facts* and *goals*. While facts are, by definition, inherently true, goals have to be achieved. Specifically, causality, in the timeline-based approach, is defined by means of a set o *rules* indicating how to achieve goals. Formally,

Definition 6. A rule is an expression of the form

$$n(x_0,\ldots,x_k) @ [s,e,\tau] \leftarrow \mathbf{r}$$

where:

- $n(x_0, \ldots, x_k) @ [s, e, \tau]$ is the head of the rule, i.e., an expression in which n is a predicate name, x_0, \ldots, x_k are constants, numeric variables or object variables, s and e are temporal variables belonging to \mathbb{T} such that $s \leq e$ and τ is an object variable representing the timeline on which the token apply.
- r is the body of the rule (or the requirement), i.e., either a slave token, a constraint among tokens (possibly including the $x_0 \ldots x_k$ variables), a conjunction of requirements or a disjunction of requirements.

Rules define causal relations that must be complied to in order for a given goal to be achieved. For each goal having the form of the head of a rule, the body of the rule must also be present in the token network. As an example, the expression $\{st_0() @[s] \leftarrow \{st_1() @[s_1] \land 10 \leq s - s_1 \leq 20\}\}$ represents a rule asserting that, for each stimulus st_0 there must exist, from 10 to 20 seconds before, a stimulus st_1 .

We have now all the ingredients to define a timeline-based planning problem. In particular, the definition can rely on the above concept of requirement.

Definition 7. A timeline-based planning problem is a triple $\mathcal{P} = (\mathcal{T}, \mathcal{R}, \mathbf{r})$, where:

- \mathcal{T} is a set of timelines.
- \mathcal{R} is a set of rules.
- r is a requirement, i.e., either a (fact or goal) token, a constraint among tokens, a conjunction of requirements or a disjunction of requirements.

The role of a timeline-based solver consists, basically, in applying the proper rules to achieve all the goals of the problem while maintaining the constraint network consistent. It is worth highlighting that, in general, the application of the rules might result in the introduction of further goals into the token network. Such goals, also called sub-goals, require to be achieved as well. The process ends up when, for all the goals of the token network or it is recognized as semantically equivalent to another token (in this case we talk about *unification* of the tokens). Notice that, despite the simplicity of the above solving procedure, the combination of disjunctions and constraints in the rules make the resolution process, in generally, extremely challenging from a computational point of view. For this reason, indeed, heuristics are often used to make the resolution process more efficient (see, for example, [7, 8]).

5 Modeling Students and Lessons through Timelines

As already mentioned, the ExPLORAA system is composed of different functional blocks. In particular, the user modeling module aims at creating and dynamically maintaining an updated model of the users which is used as a starting point for the personalization of the lessons. By pursuing the overall objective of enhancing the learning experience, indeed, it is necessary to keep a user model up-to-date in order to consider how their emotional, psychological, physiological and geographical parameters can influence the learning process. Specifically, the student modeling has three main objectives:

- Model and monitor relevant factors through which the lesson can be customized;
- Develop a model that can represent the user's profile;
- Provide a high level guidance for customizing learning objectives.

The set of considered relevant factors include, among the other things, the health status, the fatigue, the personal interests, the level of engagement and the current performance assessment. The use of Bluetooth bracelets (e.g., the Empatica $E4^{3}$), for example, allows the extraction of physiological values such as peripheral skin temperature, skin conductance, heart rate and heart variability. Additionally, it is possible to leverage on geo-localization services to get a good estimate of the users' position in time. The initial evaluation of these variables, used as a baseline to initialize the didactic experience and as a reference point for subsequent measurements, can be done through the use of standardized questionnaires or physiological measurements performed off-line before the lesson. It is worth to notice, however, that the profile of a student can also be updated exploiting the interactions of the users with the system asking them, for example, to answer to sporadic questions. As an example, the users' *engagement* is measured through a five levels Likert-type scale which is administered to the users at regular intervals. Finally, particular emphasis is given to the students' *performance* which is monitored and observed through the administration of questions and the interpretation of the provided answers.

By processing the above information, the system generates a user model that is constantly updated to perceive and represent significant changes in the emotional state (note that parameters can generally change over time). In addition, students' performance is analyzed and processed in order to gather further usable information useful for a finer customization of the lesson. Furthermore, the teacher can access this information in order to supervise and control the customization. For this purpose, this component can provide guidance on how to customize the lesson. Personalization of a training course can therefore be done automatically, but it can also be suggested to a teacher who independently decides whether to adapt the training course (i.e., according to a mixed-initiative style).

5.1 Modeling the Lessons

The modeling of the lessons is the key feature of the ExPLORAA system since it creates and manages the *network of stimuli* that guides the entire learning session. Nodes on this network are tokens and are intended to represent temporally annotated stimuli (e.g., videos, text messages, questions, etc.) to be sent,

³ https://www.empatica.com/research/e4

at appropriate time, to the users while edges represent causal and temporal relations among such stimuli introduced either in the planning problem definition or through the application of the rules. Additionally, tokens are endowed with additional information including a set of covered topics (e.g., "art", "architecture", "religion", etc.) and some content dependent on the nature of the stimulus (e.g., a text for textual stimulus or a URL for a video stimulus). Specifically, when a user enrolls to a lesson, defines which of the topics covered by the lesson are of interest to her/him. When sending stimuli to the users, a filtering procedure will guarantee the delivery of only those stimuli for which the user has declared an interest.

It is worth noting that although the above network is initialized in order to represent an abstract blueprint of a lesson, it is afterwards customized and dynamically adapted to the profile of the involved user. Personalization, indeed, takes place both in terms of users' interest in some topics, as explained earlier, as well as in terms of the type (and the number) of tokens in the token network. Specifically, adaptations to the network are made thanks to the application of a set of rules (introduced in Definition 6) associated to each lesson, which define how to "react" to the users' profile, to their updates and to their actions (e.g., moving to a specific location or answering to a question). Such rules, in particular, are intended to create the "conditions", in terms of events and their relations within the network, for other events to be present. An example of rule can be "in order to stimulate the cognitive activity of the group, either propose a simple crosswords and the group's performance is low, or propose a complex crosswords and the group's performance is high". Notice that by taking advantage of the possibility of defining disjunctions within the rules and being able to combine such rules sequentially, it is possible to obtain a great wealth of possible lessons' evolutions. Finally, since some of these rules may contain conditions which concern the user model, not all of them are applicable (e.g., in the above example, in case the group's current performance is low, only the simple crosswords is proposed), resulting in an overall network which is always compatible with the current users' profiles.

Broadly speaking, the teacher loads the chosen lesson from a database resulting in the construction of an initial token network corresponding to a set of stimuli, positioned over time, which will be communicated to users like videos, text messages, questions, etc. The network is, through the application of the rules, afterwards customized to the users participating in the lessons. By executing the lesson, then, tokens, representing stimuli and interaction requests, are dispatched, at proper time, to the users interested to them. It is worth noticing that in order to foster interaction and collaboration among them, the distributed information may be partial, requiring users the need to send messages to other students so that they can build an overview and respond appropriately to the challenges posed by the system, thus fostering cooperation. Whenever the profile of a user changes because, for example, her/his level of fatigue increases, specific rules are applied resulting in an update of the network so as to bring it back to be "consistent" with the current status of the users. Similarly, updates might happen as a consequence of users' actions, resulting in a network which is always representative of the initial lesson while being dynamically adapted to the specific context.



Fig. 3. An example of ExPLORAA dynamic lesson adaptation in case either action a_0 or action a_1 is performed by a user.

As an example, Figure 3 shows, at its top, an initial network containing three stimuli (i.e., st_0 , st_1 and st_3) and an interaction request (i.e., ir_2) representing, for example, a question. Each event has its own execution time and its covered topics (e.g., st_0 will be dispatched at 08:20). In the figure, the arrows represent the causal relations among the tokens that emerge from the application of the rules. In other words, the st_1 token is in the network "because of" the st_0 goal. Additionally, the st_1 token is constrained to be from 15 to 40 seconds before the st_0 goal. Notice that in order to simplify the explanation and make the speech clearer we have omitted the information regarding the other temporal constraints which, however, can be considered embedded within the arrows.

At 08:05, the interaction request ir_2 requires the interested users (i.e., those whose interests match the topics of the stimulus associated to the ir_2 token) to take some action (i.e., either a_0 or a_1). Suppose, as an example, that the action a_0 is chosen, the network is adapted to the one on the left by adding the new token st_4 at 08:30. Again, the st_4 token is added within the network "because of" the a_0 action. Conversely, in case action a_1 is chosen, the network is adapted to the one on the right by adding the new tokens st_5 , at time 08:35, and st_6 at 08:38. As already mentioned, it is worth noticing that this type of adaptation occurs in a similar way as a consequence of the actions carried out by the users as well as for the dynamic changes that occur to their profile, ensuring a discrete availability of flexibility and adaptation to the particular conditions that may arise in the different lessons.

6 Supporting Lesson Adaptation

An interesting feature of the proposed system regards the possibility for adapting the lessons thanks to the supported temporal flexibility. In particular, the constraint network introduced in Definition 3 manages temporal variables and constraints through a Simple Temporal Network (STN) (see [10]). Specifically, each constraint between two temporal variables x_i and x_j is limited to taking the form $l \leq x_i - x_i \leq u$ where l and u are two constants representing, respectively, the lower and the upper bound on the amount of time that lasts between the two time points x_i and x_j . The application of the rules during the resolution procedure requires the dynamic introduction of further variables, for each of the slave tokens, and additional constraints into the constraint network. For this reason we use incremental algorithms (see, for example, [16]) which are able to efficiently manage the propagation of such constraints. These algorithms are able to maintain the temporal flexibility of the involved variables, producing, for each variable x, a couple of bounds $[l_x, u_x]$ representing its lower and upper allowed values. Dechter et al., in [10], demonstrate that assigning to each variable x its earliest possible time l_x results in a consistent temporal network⁴.

While in a standard lesson execution we would use these values to establish the timing for dispatching the stimuli, problems arise when the teachers want to control their position in time. Suppose, for example, we have an STN with two variables x_0 and x_1 , representing the time points of two tokens associated to two stimuli st_0 and st_1 . Additionally, the $10 \leq x_1 - x_0 \leq 20$ constraint imposes a temporal distance between them which goes from 10 to 20 time units (although the system uses milliseconds, we maintain in the dissertation a generic measurement unit for sake of simplicity). The initial bounds of the x_0 variable is [10, 30] while the initial bounds of the x_1 variable is [20, 50]. Since the earliest possible times represent a solution for such a constraint network, st_0 and st_1 are going to be dispatched, respectively, at time 10 and at time 20. Suppose, for any didactic reason, the teacher wants to delay the execution time of the stimulus st_0 of 15 time units, a temporal constraint would reduce the bounds of the x_0 variable to [25, 30]. The propagation of the constraints would tighten the bound of the x_1 variable as well to, possibly (notice that other constraints might also be involved), [35, 50], hence right-shifting the dispatching time of st_1 at 35. At a second step, however, the teacher recognizes she/he has moved the stimulus st_1 too far and decides it is better to move it a little back. The lower bound for the st_1 stimulus, however, is now 35 and cannot be moved back through temporal constraints without introducing an inconsistency in the constraint network.

We can get around this problem by introducing, for each variable x, together with its lower (i.e., l_x) and upper (i.e., u_x) bounds resulting from the propagation of the temporal constraints, a third value d_x representing its dispatching time, whose allowed values are always maintained between the bounds of the variable (i.e., the relation $l_x \leq d_x \leq u_x$ always holds). Whenever the user decides to

⁴ It is worth noticing that the same result holds also when choosing for each variable their latest possible times.

change the position in time of a stimulus, however, only the dispatching times are updated, without touching the bounds of any variable.

In order to implement such a procedure we need, first of all, for each temporal variable x, a list of *watching* constraints *watchs* [x] (i.e., for each variable, the list of constraint watching for the variable's updates). Specifically, when a new constraint c is introduced into the constraint network, c is added to the watching lists *watchs* $[x_i]$ of each variable $x_i \in scp(c)$ belonging to the scope of c. Additionally, we need a propagation queue $prop_q$ responsible for maintaining the list of variables whose dispatching time has been updated. Whenever the dispatching time of a variable is updated, the variable is enqueued into the propagation queue.

Algorithm 1 The overall propagation procedure
procedure propagate
while $prop_q \neq \emptyset$ do
$x \leftarrow prop_q.dequeue$
for all $c \in watches [x]$ do $c.PROPAGATE(x)$

The propagation procedure is described in the Algorithm 1. Specifically, while the propagation queue is not empty, a variable x is dequeued from the queue and a propagation procedure is called for each constraint c in the watching list associated to the variable x (i.e., for all the constraints having the variable xin their scope). Additionally, for each temporal constraint c having the form $l \leq x_j - x_i \leq u$ the propagation procedure is described in the Algorithm 2. Specifically, whenever the dispatching time of the x_i (x_j) variable is updated, the propagation procedure checks whether the x_j (x_i) variable's dispatching time needs to be updated as well. In case it needs, the x_j (x_i) variable's dispatching time is updated and the x_j (x_i) variable is enqueued in the prop_q propagation queue.

The last aspect to be considered regards the initial introduction of the variables within the propagation queue. There are basically two reasons which might lead to this: either (a) a bound of a variable x is updated, within the STN propagation procedure, in a way such that the expression $l_x \leq d_x \leq u_x$ is violated or (b) a user manually updates the dispatching time of a variable x associated to a stimulus. In the first case, described in the Algorithm 3, the dispatching time is updated so as to restore the validity of the $l_x \leq d_x \leq u_x$ relation and the variable is enqueued in $prop_q$. The second case, described in the Algorithm 4, more straightforwardly, directly updates the dispatching time and enqueues the varible in $prop_q$. Notice that, after calling both the STN propagation procedure and the SET_DSPATCHING_TIME procedure, the PROPAGATE procedure (Algorithm 1) must be called in order to update the dispatching times of the related variables.

Finally, it is worth noticing that none of the above procedures is supposed to fail. The consistency checking of the constraint network, indeed, is still delegated to the STN propagation procedure.

Algorithm 2 The propagation procedure for the temporal constraint

 $\begin{array}{l} \textbf{procedure PROPAGATE}(x) \\ \textbf{if } x == x_i \textbf{ then} \\ \textbf{if } d_{x_j} < d_{x_i} + l \textbf{ then} \\ d_{x_j} \leftarrow d_{x_i} + l \\ prop_q \leftarrow prop_q \cup \{x_j\} \\ \textbf{else if } d_{x_j} > d_{x_i} + u \textbf{ then} \\ d_{x_j} \leftarrow d_{x_i} + u \\ prop_q \leftarrow prop_q \cup \{x_j\} \\ \textbf{else if } x == x_j \textbf{ then} \\ \textbf{if } d_{x_i} < d_{x_j} - u \textbf{ then} \\ d_{x_i} \leftarrow d_{x_j} - u \\ prop_q \leftarrow prop_q \cup \{x_i\} \\ \textbf{else if } d_{x_i} > d_{x_j} - l \\ \textbf{then} \\ d_{x_i} \leftarrow d_{x_j} - l \\ prop_q \leftarrow prop_q \cup \{x_i\} \\ \end{array}$

Algorithm 3 The bound update procedure called by the STN propagation procedure

procedure SET_BOUND(x, l, u) $l_x \leftarrow l, u_x \leftarrow u$ **if** $d_x < l_x$ **then** $d_x \leftarrow l_x$ $prop_q \leftarrow prop_q \cup \{x\}$ **else if** $d_x > u_x$ **then** $d_x \leftarrow d_x$ $prop_q \leftarrow prop_q \cup \{x\}$

Algorithm 4 The dispatching time update procedure
procedure set_dispatching_time(x, d)
$\mathbf{if} \ d_x \neq d \ \mathbf{then}$
$d_x \leftarrow d$
$prop_q \leftarrow prop_q \cup \{x\}$

7 The ExPLoRAA Prototype and Preliminary Evaluation

By relying on the concept sketched in Figure 2, we have realized a first prototype of the ExPLORAA ITS. In particular, several instances of the Desktop and Mobile applications, depicted in Figure 4, allow students and teachers the access to the ExPLORAA system through the remote interaction with a centralized back-end. Specifically, teachers have the opportunity to create and manage lessons while monitoring the students following them. Students, on the other hand, enroll to the available lessons specifying their interests and receive customized stimuli according to their psychophysiological state assessed through a combination of sensors and targeted questions.



Fig. 4. The ExPLORAA Desktop and Mobile Graphical User Interfaces.

We have created a lesson model example inspired by the "Madonnelle Stradaiole" of Rome. These are present in large numbers on the walls or corners of historic buildings. The idea consists in sending stimuli to users so as to guide them to visit such shrines, customizing the path to their psychophysiological state. By exploiting georeferencing, the ExPLORAA system asks users to take pictures of the shrines to be used, afterwards, to build games which, played together, reduce social isolation and stimulate cognitive activity. The first step for the implementation of a scenario of this kind consists in collecting information about the shrines like, for example, a brief history about each of them and their GPS coordinates. Additionally, a customized questionnaire, administered to users, is intended to extract an initial profile to be used as a baseline. Starting by the initial profile, a path, compatible with the profile, is selected and, step by step and at proper time, suggested to the user (e.g., "go to pos1", "the shrine at pos1 has been built in 1796", "take a picture at the shrine in pos1", "go to pos4", etc.). By taking into account physiological data, the system can adapt the route switching, for example, once the ExPLORAA system realizes that the

user is not too tired, to a longer one, fostering physical activity and a prolonged interaction with the other involved users. Finally, once back from the trip, the ExPLORAA system builds a "memory" game, challenging the participants to discover, in the fewest possible steps, pairs of the taken pictures hidden under the tiles, hence stimulating cognitive activity.

Feedback gathering has taken place at the Televita facilities. Specifically, a focus group has been held in which volunteers of the association, as well as representatives of other associations that deal with assistance to the elderly and/or psychological distress present on the territory of the III Municipality of Rome, have participated. Specifically, 6 people dealing with Televita, 3 representatives of "Società San Vincenzo de Paoli", and 1 representative of the volunteering organization "Oltre Le Barriere". The meeting (see Figure 5) has been structured around the presentation of the developed prototype that proved to be functional for eliciting comments and suggestions from the participants. We wanted to address the discussion in the direction of feedback collection for what concerns the improvement of the interface, the modalities of providing the service, the ways to enhance the engagement, as well as the possibility for the participants to provide suggestions and free comments.



Fig. 5. Focus group participants.

In general, the feedback obtained on the platform and its services was decidedly positive and the participants found the play aspect of the project particularly interesting. In fact, the game was read not only as a recreative tool, but also as a learning and social cohesion one, a crucial aspect to promote well-being in the elderly. Furthermore, the value of the possibility offered to citizens who can participate in the co-management of the public good through the municipal volunteering has emerged and, thanks to ExPLORAA, they have the opportunity to take on some aspects in a territorial area thus favoring a certain sense of belonging. This reflects the need to motivate older people to feel active, rather than a burden for society or for the family. Participation in activities motivates people and makes them feel alive. Determinant is also trying to cope with one of the biggest problems of older people: the suffering of loneliness. It emerged, indeed, that the proposed formula could represent an effective tool, especially considering the social character of the activities that are essentially based on the aggregation and comparison between different people in a game context. The possibility of personalizing the experience was judged to be very useful, both to encourage the motivation and involvement of people during the activity as well as to provide access to a wider range of users, taking into account possible physical limitations. In this sense, the personalization of services is not defined only by psicological aspects, such as the level of engagement, attention, emotional state, but also by more purely physical factors, such as fatigue. According to the participants, these characteristics make the proposal even more effective and desirable. Finally, a very interesting aspect that emerged from the interaction with the volunteers concerns the possibility of using the ExPLORAA system to favor inter-generational exchange. In fact, much emphasis has been given to the possibility of exploiting the system to make elderly and young people interact and to promote the comparison. In fact, a system focused on aspects such as sharing and group activities favors the exchange, while taking into account individual preferences and needs, thanks to the possibility of personalizing the experience.

8 Conclusions and Future Works

By pursuing their didactic needs, teachers want flexibility in adapting the resulting network. Furthermore, teachers are human beings and, as such, might change their idea. This paper introduces the ExPLORAA system as an AIbased learning environment specialized so as to support active aging. We have introduced the general idea of supporting older people in maintaining themselves mentally and physically active being helped by some intelligent technology both during a class and during excursions. The ICT intelligent core makes use of a specific kind of artificial intelligence technology called automated planning which is responsible for representing the key ingredients of the ExPLORAA system, creating a baseline "lesson" and dynamically adapting it to actions and profile updates so as to integrate both personalized stimuli and requests to the users over time. Additionally, the proposed system guarantees an efficient management of the temporal constraints and, thanks to the introduction of the proposed algorithm, the possibility for the teachers to efficiently change their minds regarding decisions made on temporal aspects. Although the application is currently under development, the constant contact with Televita's volunteers is allowing the transition from a lab prototype to an incrementally more robust version of the system to be tested in realistic scenarios.

References

 Allen, J.F.: Maintaining Knowledge about Temporal Intervals. Commun. ACM 26(11), 832–843 (1983)

- Bacon, L., Cesta, A., Coraci, L., Cortellessa, G., De Benedictis, R., Grilli, S., Polutnik, J., Strickland, K.: Training crisis managers with PANDORA. In: Proceedings of the 20th European Conference on Artificial Intelligence. pp. 1001–1002. IOS Press (2012)
- Cesta, A., Cortellessa, G., Benedictis, R.D.: Training for Crisis Decision Making - An Approach Based on Plan Adaptation. Knowledge-Based Systems 58, 98–112 (2014)
- 4. Cesta, A., Cortellessa, G., Benedictis, R.D.: Using Training with Older People for Active Ageing in Time and Space. In: AI*AAL@AI*IA (2017)
- Cesta, A., Cortellessa, G., De Benedictis, R., Fracasso, F.: A tool for managing elderly volunteering activities in small organizations. In: Esposito, F., Basili, R., Ferilli, S., Lisi, F.A. (eds.) AI*IA 2017: Advances in Artificial Intelligence (2017)
- Cortellessa, G., Fracasso, F., Sorrentino, A., Orlandini, A., Bernardi, G., Coraci, L., De Benedictis, R., Cesta, A.: ROBIN, a Telepresence Robot to Support Older Users Monitoring and Social Inclusion: Development and Evaluation. Telemedicine and e-Health (2017)
- De Benedictis, R., Cesta, A.: Integrating Logic and Constraint Reasoning in a Timeline-based Planner. In: AI*IA 2015 - XIVth International Conference of the Italian Association for Artificial Intelligence (2015)
- De Benedictis, R., Cesta, A.: New Heuristics for Timeline-Based Planning. In: Proceedings of the 6th Italian Workshop on Planning and Scheduling A workshop of the XIV International Conference of the Italian Association for Artificial Intelligence (AI*IA 2015). pp. 33–48 (2015)
- 9. Dechter, R.: Constraint Processing. Elsevier Morgan Kaufmann (2003)
- Dechter, R., Meiri, I., Pearl, J.: Temporal Constraint Networks. Artificial Intelligence 49(1-3), 61–95 (May 1991), http://dx.doi.org/10.1016/0004-3702(91)90006-6
- Ghallab, M., Nau, D., Traverso, P.: Automated Planning: Theory and Practice. Morgan Kaufmann Publishers Inc. (2004)
- Gronlund, N.E., Brookhart, S.M.: Gronlund's Writing Instructional Objectives. Pearson/Merrill Prentice Hall (2009)
- 13. Lecoutre, C.: Constraint Networks: Techniques and Algorithms. Wiley-IEEE Press (2009)
- Muscettola, N.: HSTS: Integrating Planning and Scheduling. In: Zweben, M. and Fox, M.S. (ed.) Intelligent Scheduling. Morgan Kauffmann (1994)
- Ohlsson, S.: Some principles of intelligent tutoring. Instructional Science 14(3), 293–326 (May 1986), https://doi.org/10.1007/BF00051825
- 16. Planken, L.R., de Weerdt, M.M., Yorke-Smith, N.: Incrementally Solving STNs by Enforcing Partial Path Consistency. In: Hoffmann, J., Kautz, H., Brafman, R., Geffner, H. (eds.) Proceedings of the Twentieth International Conference on Automated Planning and Scheduling (ICAPS-10). pp. 129–136. AAAI Press, Menlo Park, CA, USA (May 2010), http://repository.tudelft.nl/view/ir/uuid:3ca1a60e-8de4-4207-93aa-27e1442ecc18/
- VanLehn, K.: The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems. Educational Psychologist 46(4), 197–221 (2011)
- Wenger, E.: Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1987)