

Applying Case-Based Planning to Personalized E-learning

Antonio Garrido
Universitat Politècnica de València
Camino de Vera s/n, 46022
Valencia (Spain)
Email: agarridot@dsic.upv.es

Lluvia Morales
University of Granada
Cuesta del Hospicio s/n, 18070
Granada (Spain)
Email: lluviamorales@decsai.ugr.es

Ivan Serina
Free University of Bozen-Bolzano
viale Ratisbona 16, 39042
Brixen-Bressanone (Italy)
Email: ivan.serina@unibz.it

Abstract—Sequencing of Learning Objects (LOs) has been an important issue in the last decades. From a technical perspective, we can take advantage of Artificial Intelligence (AI) planning techniques that allow us to adapt these sequences to pedagogical and students' requirements. However, there is neither a standard way to represent and compile such LO knowledge into a planning model, nor an optimal way to deal with changes during the execution of the previously adapted learning sequences. In this paper, we propose a general and effective approach to automatically extract information from the LOs to create planning domains, which are then solved by case-based plan merging techniques that are also used as a recommender system. This way of proceeding allows the teacher to store, and reuse, the best learning routes for each student's profile and course objectives. When discrepancies on the student's profile or state are detected during the course execution, the system assists the teacher in readapting, repairing or improving the route to fit the new objectives.

I. INTRODUCTION

E-learning is becoming a high-impact innovative topic as it offers a promising way to facilitate and enhance the learning process by combining LOs to create flexible courses. In particular, AI planning techniques have shown to be very adequate to generate fully tailored routes of LOs according to pedagogical theories [1]. More specifically, the planning objective in an e-learning setting is to provide a student-centered solution (i.e. personalized learning) by offering a flexible learning process where courses and LOs are tailored to the specific needs, learning styles, objectives, background and, in general, the profile of each student [1], [2], [3], [4].

We focus on the application of Case-Based Planning (CBP) techniques [5] for the definition, memorization, retrieval and adaptation of learning routes. However, defining a high quality learning route is difficult; it depends on many variables, involving the LOs, the available resources, the students, and their preferences/learning styles. Thus, our goal is to help teachers choose the most suitable learning route and (semi-)automatically adapt it in accordance with the students' goals and individual features. Hence, our approach contributes with:

- 1) A mapping of learning courses into planning domains by using an automated translation of e-learning templates [2] into PDDL durative actions [6].
- 2) A CBP repository for planning domain+file compilations that contains students' learning information, which are successively analyzed by our case-based planner [5].

- 3) A flexible way to make curriculum authoring easier because teachers can easily retrieve, improve and reuse courses previously generated.
- 4) A simple translation of the resulting sequence of LOs into another standard representation, namely learning design, that provides a usable manifest for on-line learning platforms, thus closing the e-learning cycle.

The paper is structured as follows. First, we introduce the role of planning for learning routes personalization, together with some basic background on planning and CBP techniques. Next, we present our general approach, describing its structure and main elements: i) required metadata information, ii) compilation of this metadata into a PDDL description, iii) planning personalized learning routes, iv) integration with Learning Management Systems (LMSs), and v) navigation and monitoring of the routes. Finally, we discuss the lessons learnt and raise the most important issues that are necessary for a full support of planning within an e-learning setting.

II. THE ROLE OF AI PLANNING IN PERSONALIZED E-LEARNING

A. Using planning for personalization

The traditional mode of instruction (one-to-many lecturing, or one-to-one tutoring), which is adopted in conventional education, cannot fully accommodate the different learning and studying styles, strategies and preferences of diverse students. Consequently, offering personalized learning routes to individual students' needs and profiles is essential to promote better learning initiatives in e-learning.

Educational systems have always aimed at ensuring flexibility within course personalization. Many authors have addressed this flexibility using different techniques, such as adjacency matrices, integer programming models, neural networks, graph-based sequencing procedures and, more recently, intelligent planning techniques [4], [7], [8], [9], [10], [11]. We use planning (and scheduling) techniques to bridge the gap between the purely e-learning necessities and the accommodation of temporal+resource constraints of the environment, to make the course applicable in a real scenario, which is usually missing in previous approaches. We go beyond the traditional e-learning insights and give support not only to adaptation

and LO sequencing, but also to scheduling constraints and multi-criteria optimization metrics. This raises a challenge for a successful integration with LMSs that facilitate the dynamic navigation of contents/LOs, monitor the students' progress when following their proposed learning routes, check whether some discrepancies appear and react to them to adapt the routes to the new necessities. And this monitoring part is also missing in most current approaches. In this monitoring line, [4] uses specific planning capabilities to: i) wait for specific information in critical points, ii) update the students' profile+state information, and iii) resume planning until the next critical point. Our approach also manages dynamics in e-learning, but when a discrepancy appears we use a case-based planner which, rather than creating a new route from scratch, retrieves the best element that fits the current requirements from a CBP repository and adapts it if necessary.

B. Some background on planning

AI planning is the task of finding a solution within a search space defined by the application of legal actions (grounded operators). Each action has preconditions and effects. Preconditions need to be satisfied before the action is applied, whereas effects describe the results (in terms of what changes) after the action is executed. More formally, a planning problem is defined as the tuple $\Pi = \langle \mathcal{F}, \mathcal{I}, \mathcal{G}, \mathcal{O}, \mathcal{M} \rangle$, where:

- \mathcal{F} represents the fluents of the problem, defined as ground atomic propositional formulae or numeric variables;
- \mathcal{I} represents the initial state;
- \mathcal{G} represents the goal conditions to be achieved;
- \mathcal{O} is a finite set of operators, where each operator is defined as a tuple $\langle Pre, dur, Effs \rangle$. Each precondition in Pre defines a propositional or numeric precondition that expresses a constraint as a tuple $\langle f\text{-exp1}, \text{binary-comp}, f\text{-exp2} \rangle$, where $f\text{-exp1}$ and $f\text{-exp2}$ represent functional expressions and $\text{binary-comp} \in \{<, \leq, =, >, \geq\}$ is a binary comparator. $dur \in \mathcal{R}^+$ is a positive duration. Finally, each effect in $Effs$ can be either a positive or negative propositional effect or a numeric effect as a tuple $\langle f\text{-head}, \text{assign-op}, f\text{-exp} \rangle$, where $f\text{-head}$ represents a numeric variable, and $\text{assign-op} \in \{:=, +, -, *, /, =\}$ is an assignment operator which updates the value of $f\text{-head}$ according to the functional expression $f\text{-exp}$.
- \mathcal{M} is a multi-criteria metric to be minimized/maximized, thus addressing optimal planning.

The planning goal is to come up with a proper plan, i.e. a partially ordered sequence of actions, whose execution will transform the initial state \mathcal{I} into a final state in which all goals \mathcal{G} are satisfied, while also optimizing the metric \mathcal{M} .

C. Description of CBP techniques

In CBP, previously generated plans are stored as cases in memory and can be reused to solve similar planning problems in the future. CBP can save considerable time over planning from scratch, thus offering a potential (heuristic)

mechanism for handling intractable problems. Similarly to other Case-Based Reasoning (CBR) systems, CBP is based on two assumptions on the nature of the world [12]. The first assumption is that the world is regular: similar problems have similar solutions. As a consequence, solutions for similar problems are a useful starting point for new problem-solving. The second assumption is that the types of problems an agent encounters tend to recur; hence future problems are likely to be similar to current problems.

In general, the following steps are executed when a new planning problem must be solved by a CBP system:

- 1) *Plan Retrieval* to retrieve cases from memory that are analogous to the current (*target*) problem.
- 2) *Plan Adaptation* to repair any faults found in the new plan.
- 3) *Plan Revision* to test the solution new plan π for success and repair it if a failure occurs during execution.
- 4) *Plan Storage* to eventually store π as a new case in the case base.

Following the formalization proposed by [13], a *planning case* is a pair $\langle \Pi_0, \pi_0 \rangle$, where Π_0 is a planning problem and π_0 is a plan for it, while a plan library is a set of cases $\{\langle \Pi_i, \pi_i \rangle | 1 \leq i \leq m\}$.

To the end of applying the reuse technique, it is necessary to provide a plan library from which “sufficiently similar” reuse candidates can be chosen. In this case, “sufficiently similar” means that reuse candidates have a large number of initial and goal facts in common with the new instance. However, one may also want to consider the reuse candidates that are similar to the new instance after their objects have been systematically renamed; this corresponds to identifying a mapping between the objects of the reuse candidate and the objects of the new instance such that the number of common goal facts is maximised and the additional planning effort to achieve the initial state of the plan library is minimised. This is extremely important in our context, where teachers could decide to reuse a course, or a part of a course, that has been adopted by their colleagues or by themselves previously. Obviously, the students will not be the same, but if their profile, their goals and the resources available are similar to the corresponding ones in the case base, our system will be able to propose a new high quality learning path to the teacher with a limited number of changes w.r.t. the one stored in the library. Our approach uses an approximate evaluation based on kernel functions [5] in order to compute an appropriate mapping between the students and the objects of the reuse candidate and the corresponding ones of the current instance, which can be computed in polynomial time.

The plan adaptation system consists in reusing and modifying previously generated plans to solve a new problem and overcome the limitations of planning from scratch. Moreover, any kind of planning system that works in dynamic environments has to take into account failures that may arise during plan generation and execution. In this respect, CBP is not an exception; this capability is called plan revision and it is divided into two subtasks: evaluation and repair. The

evaluation step verifies the presence of failures that may occur during plan execution when the plan does not produce the expected result. When a failure is discovered, the system may react by looking for a repair or aborting the plan.

After finding the plan from the library and after repairing it with the plan adaptation techniques, the solution plan can be inserted into the library or be discarded.

In order to improve the efficiency of the system and reuse as much as possible parts of previously executed plans we have adopted *plan merging techniques* [14], which are based on the well-known *divide and conquer* strategy. In order to apply this strategy, our system must accomplish two further subtasks: *problem decomposition* and *plan merging*. The problem decomposition is performed identifying the set of actions and the initial facts needed for a single goal and storing them in the case base as a new problem instance (if not already present); moreover these new instances remain related to the original solution plan in order to maintain a statistic of their effective usage. The stored (sub)cases are then used in the merging phase in order to identify a single global plan that satisfies all goals. We progressively identify the unsatisfied goals and the corresponding (sub)cases that allow to satisfy them, giving the preference to the (sub)plans that allow us to improve the plan metric and that have been successful in a greater number of times in analogous situations.

III. EXPLANATION OF OUR APPROACH

Figure 1 provides the overall schema of our approach, which involves several technological issues: i) use of common LO repositories and modeling tools, ii) compilation of planning domains, ii) algorithms for students' information acquisition, iii) application of CBP as solving techniques, iv) visualization of learning designs on LMSs, and v) monitoring students' progress when following the course to detect discrepancies. We now explain each of these issues in more detail.

A. LO metadata information and its use in modeling courses

Modeling an instructional course involves selecting LOs and defining how they are interrelated (i.e. their causal dependencies), which is done by the course designer. This highly relies on the metadata information defined in the LO itself. Metadata specification for LOs is usually specified in an XML standard format, such as LOM [15]. This specification has many useful entries for pedagogical theories, but only a few of them are really essential to support planning personalization. First, we need the technical platform requirements, seen as the particular resources for the LOs. Second, we need the educational information about the student's learning style (profile), difficulty of the LO and its typical learning time (i.e. duration). Finally, we need the relations as the content dependencies which comprise hierarchical structures and orderings among LOs. The hierarchical structures use the *IsPartOf* relationship, which represents an aggregation of LOs. For instance, in Figure 2 "LO4 *IsPartOf* LO1", which means that LO4 is compulsory in a learning route that includes the (higher level) LO1. There are also three types of ordering

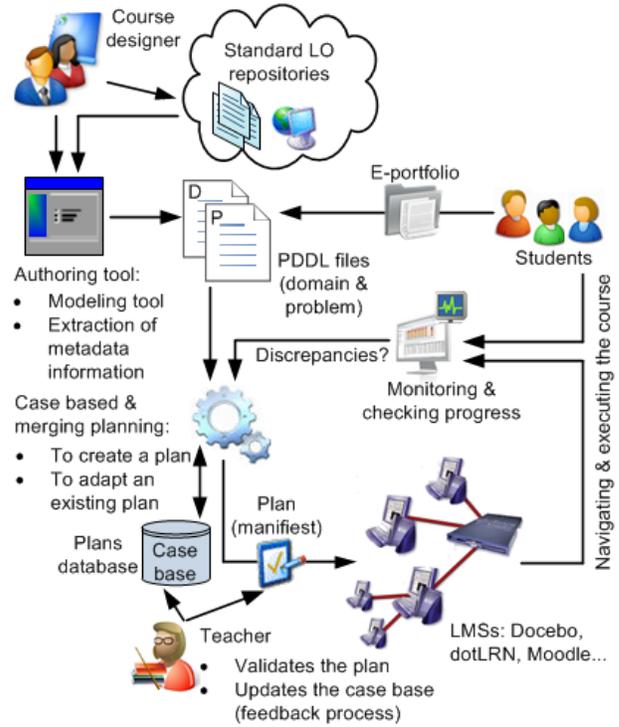


Figure 1. Overall structure of the system

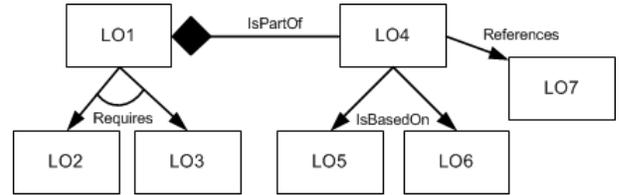


Figure 2. An example of LOs that use the LOM relations

relations to represent causal dependencies: i) *Requires*, as a conjunctive precondition; ii) *IsBasedOn*, as a disjunctive precondition; and iii) *References*, as a (soft) recommended precondition that may involve a kind of incentive or learning reward. In Figure 2, both "LO2 and LO3" are required by LO1 but only "LO5 or LO6" are required by LO4. On the contrary, the *References* relation does not denote a hard precondition but a soft one to complete a LO before proceeding with the next one (e.g. LO7 in Figure 2).

B. Planning compilation: from course+students' information to planning

The planning compilation is a knowledge-engineering task that is automated by using a mapping between e-learning metadata information and planning elements. This means to express LOs as PDDL operators and to model the students' profile and learning goals as part of a PDDL problem.

The general algorithm to compile a planning domain is to iterate all over the LOs and generate one PDDL planning operator per LO, as detailed in the mapping of Table I. This generation relies on a closed world assumption, and if new

LOs are to be used the domain must be recompiled. Anyway, this compilation is very efficient, as each operator comprises five entries which are automatically extracted from the values of the LO metadata specification:

- A unique name taken from the LO name.
- One parameterized student, which facilitates the application of the same operator to different students.
- The LO duration (*typicalLearningTime* field) to model a PDDL durative action [6].
- The preconditions to support all the dependency relations, according to the semantics of conjunctive (*Requires* and *IsPartOf*), and disjunctive (*IsBasedOn*) preconditions. A dummy precondition (`not (action-name ?s done)`) is used to avoid planning the same action more than once. Furthermore, other educational requirements, such as the intended student's role, his/her learning style, the difficulty of the LO, the required language for the LO or multimedia requirements can be modeled both as strong (*Requires* or *IsBasedOn*) or soft (*References*) preconditions.
- The effects to represent the LO outcome, i.e. having it done. They also include a `reward` to offer a full support for LO adaptation to the students, that is, how the learning resource type of each LO fits the student's learning style, based on Honey-Alonso's, Felder's or any other classification. Optionally, the compilation can include numeric expressions or resource costs, as are common in planning and scheduling, and even a particular sequencing used to assist hierarchical decomposition.

According to this compilation, the PDDL durative action for LO4 of Figure 2 is:

```
(:durative-action LO4
:parameters (?s - student)
:duration (= ?duration 5) ;the typicalLearningTime
:condition (and
  (at start (not (LO4 ?s done)))
  (or
    (at start (LO5 ?s done))
    (at start (LO6 ?s done)))
  (or
    (at start (LO7 ?s done))
    (at start (true))))
:effect (and
  (at end (LO4 ?s done))
  (at end (increase (reward)
    (reward-value ?s Equipment))))))
```

After the planning domain is generated, the planning problem is compiled by extracting the relevant student's characteristics from his/her e-portfolio. The planning problem includes the initial state, the goals and, if needed, the metric to optimize. The initial state represents the students' profile and background, the language of the course, and some other information (e.g. special equipment or resource availability). The goal is to pass the entire course or a part of it. An example of the planning problem for a `student1` that wants to attain both LO1 and LO5 outcomes in the shortest time is:

```
(:init
;Profile and learning style
  (learning-style student1 theoretical)
  (performance student1 high)
;Academic trajectory (background)
  (language-level es student1 high)
  (language-level en student1 high)
  (availability student1 much))
(:goal (and
  (LO1 student1 done)
  (LO5 student1 done)))
(:metric minimize (total-time)) ;shortest time plan
```

C. Application of plan merging techniques

Case-based plan merging techniques have been used in our approach to store learning routes (plans) previously validated by the teachers in the case base. These solution plans can be manually generated by the teachers or simply generated by a domain independent planner such as LPG, SGPlan or by our case-based planner itself, and then validated by the teachers. Moreover, in order to reuse as much as possible parts of previously executed plans, we decompose the solution plans into subparts that allow us to satisfy every single goal and we store them in the case base, if they are not already present.

When a new e-learning planning problem must be solved, we search in the case base if a plan exists that already solves all goals. If such a plan does not exist we apply, as previously exposed, plan merging techniques that progressively identify (sub)plans in the case base that can satisfy the goals. Intuitively, the adaptation consists in reusing parts of the retrieved plans to complete a new one that is similar to the original one. This allows the teachers to easily validate the proposed learning route since they can simply consider and analyze the parts of the learning route that differ from the elements stored in the case base and that have been introduced in order to satisfy, for example, new users' goals or prerequisites of LOs that do not fit with students' requirements, instead of reconsidering the whole learning route. Note that different criteria can guide the definition of a learning route. In our current version we do not only try to find good quality plans that best fit the students' requirements but also to minimize the number of LOs that have been introduced or removed w.r.t. the case base elements. The relative importance of plan quality w.r.t. plan stability can be chosen by the teacher when the case-based planner is executed.

D. Closing the cycle

The global solution plan is then provided to the LMS that identifies, for each student, the instructional design that must be followed by the students under the IMS-CP or SCORM specifications. Moreover, all of these (sub)plans or instructional designs, in addition to those plans of the same topic that can be recommended by the teachers, are additionally stored in the case base, if not already present.

Note that the LMS is fundamental not only to provide the students a sequence of LOs but also to visualize the learning plan to the teachers and allow them to monitor the students' progress, to easily detect significative discrepancies between the current situation and the scheduled activities, to select a single LO or sequences of LOs that can be integrated in

LO metadata item	→	PDDL action entry
Identifier		action-name
-		:parameters (?s - student), to model the student
typicalLearningTime		:duration for temporal planning
dependency relations: type of relation: case (<i>Requires, IsPartOf</i>): conjunctive (and) case (<i>IsBasedOn</i>): disjunctive (or) case (<i>References</i>): recommendation the LO: entry		:condition (and (at start (not (action-name ?s done)) if and-precondition: (and ... else-if or-precondition: (or ... for each entry: (at start (action-entry ?s done) ... other optional preconditions for current profile)
model that the action has been done, and increase a reward/utility expression due to learningResourceType (profile adaptation) and the References relation		:effect (and (action-name ?s done), and (increase (reward ?s) (reward-value ?s LRT) ... other optional rewards and/or costs)

Table I
GENERAL LO METADATA MAPPING TO PDDL ACTIONS. IRRELEVANT INFORMATION HAS BEEN IGNORED.

the case base, to manually modify the LOs proposed by the planner and, finally, to validate the instructional design the students have to perform. Furthermore, in order to easily define a new course, the LMS allows the teachers to import whole courses from external repositories; their structure with the relations among their LOs and the corresponding pedagogical motivations are directly stored in the case base, while the LOs are added to the domain description allowing the planner to effectively identify their causal structure and reuse them in different parts of the course.

While the students are interacting with the LMS and performing the previously planned LOs, some changes can occur in his/her profile information, either by external or internal conditions. Some examples of external conditions are:

- The student has taken classroom courses on different languages and increased his/her language level.
- Some prerequisites of LOs of his/her instructional design have been satisfied in other courses.
- The student has new temporal constraints (a new job or gets sick), and now (s)he has less time to accomplish the goals of the course, being unable to perform some activities.
- The student has bought a new computer that has more equipment capabilities than the previous one.
- Some equipment is no longer available, for example the student's computer is broken or a laboratory is busy.

The changes in students' profiles caused by external conditions must be modified directly by the students themselves, or by the teachers using the LMS interface for this purpose. On the other hand, examples of some internal conditions are:

- During the course execution, some tests or questionnaires are applied to evaluate the students' comprehension of the objectives. If this comprehension is negatively evaluated by getting a low score, then it is inferred that the students' performance is decreasing.
- When it is detected that the student uses LOs with a recurrent learning resource type, this behavior indicates that his/her learning style orientation is changing.

These conditions can conventionally and automatically

change the students' profile through a set of pedagogical rules. These rules can also be statically defined into the database as default rules that can be modified by the tutor (by using an intuitive interface).

When changes in the student's profile are detected, we simulate the execution of the remaining part of the learning plan in order to identify if it contains flaws, i.e. if the preconditions of LOs and the goals are no longer satisfied (this can be computed efficiently in polynomial time w.r.t. the number of actions in the remaining part of the plan). If an inconsistency is detected, it is highlighted to the teacher and (s)he can decide whether to repair it manually or to ask for a new plan to the case-based planner. If no inconsistency is detected, a new schedule of the remaining LOs is provided to the student in order to better satisfy his/her requirements and time availabilities; note that this new schedule can be simply computed in polynomial time w.r.t. the number of LOs and resource involved, and does not require any kind of validation by the teacher. Moreover, the student and the teacher can also ask the planner if a new plan of better quality, according to the new student's profile and the current resources, can be found. Anyway, when a new plan is computed by our system it must be always validated by the teacher before its execution. And the plan stability w.r.t. previously executed plans is of capital importance to reduce the teacher's overhead.

When the plan execution finishes and all the students' goals are satisfied, the corresponding plan is provided to the case-based system in order to be stored in the case base, if not already present, closing in this way the learning cycle.

IV. DISCUSSION AND FUTURE WORK

Considering the huge number of LO repositories and courses, implemented in different XML standards such as SCORM, LOM or IMS, available on the Web, it is a real challenge to offer a general approach to deal with personalized e-learning routes that fit the students' profiles and styles in a flexible way. But given the strong resemblance between planning and the creation of learning routes, we can apply planning and scheduling techniques to deal with i) course

designs, ii) LOs and learning activities sequencing, and iii) adaptation to students' profiles together with pedagogical theories. By defining a versatile mapping (such as the one presented in Table I), we can automatically reason on the time a LO requires, its difficulty, the whole duration of the course, the available time each student has, his/her preferences and learning styles, the resources that are available, and even the level of cooperation among tutors and students. This provides many opportunities to support e-learning personalization and routes adaptation.

Previous features face particular aspects of the whole e-learning picture, but there are other aspects related to the LO metadata specification, course modeling, the role of the teacher once a learning route has been created and when this route is eventually executed by the student. All in all, what we have learnt is that the following issues are important in personalized e-learning settings and can fill, in themselves, their own research lines:

- LO metadata focuses on the specification of educational and pedagogical matters, but important entries to promote personalization are still missing; e.g. ontologies on the use of the relations, use of resources, collaboration on the use of a LO, and definition of temporal constraints.
- Planning techniques need a well-defined domain and problem specification, which is not always easy to generate. We have provided here a general, flexible mapping, but some aspects are still a bit complex to extract from the metadata description. For instance, it is not intuitive how to measure the incentive value, i.e. reward, when a recommendation (soft precondition) holds.
- Teachers are not always *happy* with the learning routes that are automatically generated and feel reluctant to abandon their traditional role of *human planners* when creating learning routes. Therefore, a mixed-initiative approach that allows teachers to make further changes seems the most adequate way of proceeding.
- LMSs need to be extended to provide capabilities for integration with monitoring and checking the students' progress when executing the learning route. So, it is not only a matter of plugging in a planner to find a route, but also to provide additional tools to find out discrepancies (differences between the expected and the real student's state) to adapt the route. And here CBP techniques show very useful to adapt existing routes, at little cost on the average, rather than creating them again from scratch.
- The evaluation of this approach is not simple as it involves many parameters —we refer the interested reader to [16] for further details. But it is important to note that the success of this approach cannot be directly assessed through the students' grades/scores; we mainly aim at improving the students' satisfaction when using personalized routes, which enhances personal instruction.

Our current work focuses on the planning aspects and on the integration with Moodle LMS. On the one hand, we are working on modifying the case base repositories by including

the new plans that are progressively generated by our case-based planner [5]. Also, the teacher will be able to modify the standard LO repository, which can be now extended to be not only a repository of LOs but also a course (sequence of LOs) repository. On the other hand, we are trying to offer an integrated approach of planning and Moodle by means of Web services. Although our first attempt relies on Moodle and ILIAS, we also expect to create integration modules for other open sources LMSs, such as dotLRN and Claroline.

V. ACKNOWLEDGEMENTS

This paper was partially funded by the Consolider AT project CSD2007-0022 INGENIO 2010 of the Spanish Ministry of Science and Innovation, the MICINN project TIN2008-06701-C03-01, the Mexican National Council of Science and Technology, the Valencian Prometeo project 2008/051 and the BW5053 research project of the Free University of Bozen-Bolzano.

REFERENCES

- [1] P. Brusilovsky and J. Vassileva, "Course sequencing techniques for large-scale web-based education," *International Journal Continuing Engineering Education and Lifelong Learning*, vol. 13, no. 1/2, pp. 75–94, 2003.
- [2] A. Garrido, E. Onaindia, L. Morales, L. Castillo, F. S., and D. Borrajo, "Modeling e-learning activities in automated planning," in *Proceedings of the 3rd International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS-ICAPS 2009)*, 2009, pp. 18–27.
- [3] D. Peachy and G. McCalla, "Using planning techniques in intelligent systems," *International Journal of Man-Machine Studies*, vol. 24, pp. 77–98, 1986.
- [4] C. Ullrich and E. Melis, "Pedagogically founded courseware generation based on HTN-planning," *Expert Systems with Applications*, vol. 36, no. 5, pp. 9319–9332, 2009.
- [5] I. Serina, "Kernel functions for case-based planning," *Artificial Intelligence*, vol. 174, pp. 1369–1406, 2010.
- [6] M. Fox and D. Long, "PDDL2.1: an extension to PDDL for expressing temporal planning domains," *Journal of Artificial Intelligence Research*, vol. 20, pp. 61–124, 2003.
- [7] L. Castillo, L. Morales, A. Gonzalez-Ferrer, J. Fdez-Olivares, D. Borrajo, and E. Onaindia, "Automatic generation of temporal planning domains for e-learning," *Journal of Scheduling*, vol. 13(4), pp. 347–362, 2010.
- [8] A. Garrido and E. Onaindia, "On the application of planning and scheduling techniques to e-learning," in *Proc. 23rd Int. Conference on Industrial Engineering & Other Applications of Applied Intelligent Systems (IEA-AIE 2010) - LNAI 6096*. Springer, 2010, pp. 244–253.
- [9] N. Idris, N. Yusof, and P. Saad, "Adaptive course sequencing for personalization of learning path using neural network," *Int. J. Advance. Soft Comput. Appl.*, vol. 1(1), pp. 49–61, 2009.
- [10] G. Knolmayer, "Decision support models for composing and navigating through e-learning objects," in *Proc. Annual Hawaii Int. Conference on System Sciences (HICSS'03)*, 2003, pp. 31–41.
- [11] E. Kontopoulos, D. Vrakas, F. Kokkoras, N. Bassiliades, and I. Vlahavas, "An ontology-based planning system for e-course generation," *Expert Systems with Applications*, vol. 35(1-2), pp. 398–406, 2008.
- [12] D. Leake, *Case-Based Reasoning*. Cambridge, Massachusetts: The MIT Press, 1996.
- [13] P. Liberatore, "On the complexity of case-based planning," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 17(3), pp. 283–295, 2005.
- [14] Q. Yang, D. Nau, and J. Hendler, "Merging separately generated plans with restricted interactions," *Computational Intelligence*, vol. 8(4), pp. 648–676, 1992.
- [15] LOM, "Draft standard for learning object metadata. IEEE. rev. 16 february 2005," 2002, available at http://ltsc.ieee.org/wg12/files/IEEE_1484_12_03_d8_submitted.pdf (accessed May 2011).
- [16] L. Morales, A. Garrido, and I. Serina, "Planning and execution in a personalised e-learning setting," in *CAEPIA (in press)*, 2011.