

Planning and Execution in a Personalised E-learning Setting

Lluvia Morales¹, Antonio Garrido², and Ivan Serina³

¹ University of Granada (Spain),

² Universitat Politècnica de València (Spain),

³ Free University of Bozen-Bolzano (Italy)

lluviamorales@decsai.ugr.es, agarridot@dsic.upv.es, ivan.serina@unibz.it

Abstract. The main aim of e-learning is to provide a learning route where activities are tailored to individual necessities. But this is not always enough, as this route needs to be executed in a real learning management system where some discrepancies (between the real and expected situation) may appear. In this paper we focus on the generation of these routes from a planning perspective, but also on the monitoring and execution of the routes and, in case of significant discrepancies, provide a planning approach for adapting the route —rather than generating a new one from scratch. We demonstrate that this approach is very valuable to maximise the *stability* of the learning process, and also for the performance and quality of the learning routes.

Keywords: applications of AI, e-learning, planning, personalisation of e-learning routes

1 Introduction

E-learning is, in essence, a multidisciplinary field that takes advantage of the current advances in technology and integrates many techniques from different fields, such as educational theories, profile identification and modelling, knowledge representation, AI methods and optimisation procedures among others.

The minimal component of e-learning is a Learning Object (LO), which is an interoperable resource to be used in flexible learning routes that support and enhance learning. Thus, LOs have been likened to LEGO bricks and the way they can be stacked to form bigger structures and reused once and again. In other words, the utmost LO reusability cannot be achieved by considering the LOs as isolated components, but as aggregated elements for large courses to be eventually executed by students. From this execution perspective, we have to deal with two challenging issues. First, how to build the right sequence of LOs for each student, i.e. to provide a learning route where LOs and activities are tailored to the specific needs, objectives, background and, in general, profile of each student (personalised learning [5, 10, 14]). Second, how to monitor the execution of the learning route, check its progress and act when discrepancies (differences w.r.t. the expected state) appear, i.e. to provide a flexible adaptation

process that does not ignore the original student's interests and tries to reuse the original route as much as possible. This paper mainly builds on these two issues and contributes with an AI planning approach to: i) model and encode learning courses and students' profiles as planning problems; ii) solve these problems to find plans, i.e. learning routes, that entirely fit the students' interests; iii) monitor the execution of the learning routes checking its validity; and iv) adapt the route in the event of a discrepancy that prevents the execution of the route.

The structure of the paper is as follows. Section 2 presents some related work for course composition and personalisation of learning routes. In section 3 we propose our schema for planning e-learning routes, give a short description on the knowledge representation stage to compile the corresponding planning problem and detail the importance of stability in e-learning. How we can interleave execution, monitoring and adaptation of learning routes is deeply explained in section 4. Section 5 shows our experimental results and, finally, section 6 concludes the paper.

2 Related Work

Course composition has been traditionally seen from two perspectives: i) adaptive courseware generation, and ii) dynamic courseware generation. In the former, the idea is to sequence an individualised course taking into account specific learning goals and the student's previous knowledge. Thus, the main goal is to ensure that a student completes all the activities that an instructor deems important, which makes the objective instructor-centered [1]. Many techniques have been successfully applied to generate personalised courses as a means to bring the right content to the right person, such as adjacency matrices, integer programming models, neural networks and AI planning [3, 5, 7, 8]. In the latter, the system observes the student's progress during his/her interaction with a general course and dynamically adapts it according to the specific student's needs and requirements [11, 13]. Hence, the goal is to assist students in navigating in a complex information space in order to achieve whatever goal they choose, making this type of hypermedia technique student-centered. In other words, the adaptive generation selects LOs from a given repository in a way which is appropriate for the targeted individuals, whereas the dynamic generation provides a more accurate browsing associated with an on-line course in an optimal order, where the optimisation criterion takes into consideration the student's background and performance.

There are a few works, such as [2, 14], that combine the two previous perspectives as part of their own intelligent tutoring systems. But in most cases, once the route is created, the monitoring part to check its execution is missing. This represents an important limitation, because it is not only important to generate a personalised route, but also to check how it is navigated and executed by the student, and adapted if some discrepancies (between the real and expected situation) appear. Revisiting the LEGO metaphor, it does not suffice with having the plan of a big structure because we also need to put it into practice. And if

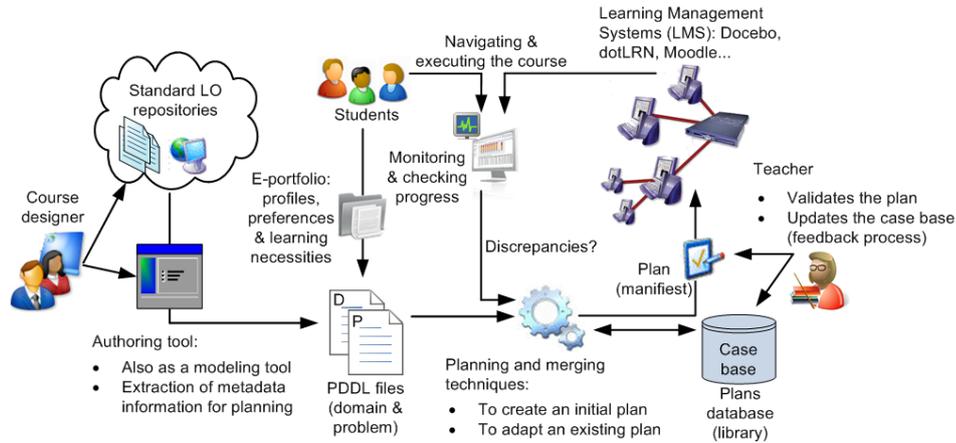


Fig. 1. Overall structure for using planning in an e-learning setting.

one brick is missing when creating the structure, we do not discard the whole structure but try to replace the missing brick with others that play a similar role (and reuse as much part of the original structure). This paper overcomes this limitation by using e-learning and planning standards, making it more general and applicable.

3 A General Approach for Planning E-learning routes

3.1 Description of the structure at a glance

Fig. 1 depicts the general structure of our approach. The idea is to define a course by using LOs available in (Web) repositories. Once the course has been defined and the students' profiles—in terms of background, learning styles and interests—have been modelled, an automatic translator compiles all this information as a planning problem to be solved by a standard planner. The planner generates a plan, i.e. a learning route, that is validated by one teacher (and also stored in a plan library). This route is uploaded to a Learning Management System (LMS) as a plan manifest that allows a student to navigate through his/her tailored route. The LMS also monitors the execution of each route and if any discrepancy is found between the real and expected state, a new planning iteration is launched to fix (adapt/replan), or improve, the learning route. Note that the new route should not significantly differ from the original one, so keeping a high stability record is important in e-learning.

3.2 Compilation of domain+planning problem

This is a knowledge representation stage and provides the foundations for using planning technology. It consists in mapping the information about the course (the

LOs, their relationships, technical and educational requirements) into PDDL actions, which define the planning domain. LOs are usually labelled by an XML metadata format, such as LOM [9]. Thus, the planning domain is generated by iterating all over the LOs of the course to generate one PDDL action per LO. This compilation is very efficient (polynomial time), as each action comprises four entries automatically extracted from the values of the LO metadata specification: i) name of the LO; ii) duration of the LO (learning time); iii) preconditions, based on the profile's dependence plus the relations defined in its metadata; and iv) effects, based on the learning outcomes. For further details about this compilation see [5].

On the other hand, the planning problem is compiled by extracting the relevant students characteristics from his/her e-portfolio, which are obtained from the XML files in IMS-LIP and IMS-QTI standards. In addition to the initial state (background) and learning goals per student, it can also include the metric to be optimised, such as finding the shortest learning route or the one that maximises a given reward, score or learning utility.

The generation of a PDDL planning problem facilitates the use of independent solvers, which provides a nice approach to abstract the e-learning specific features from the planning details. When a plan is found, as a sequence of LOs that best suits the student, it is displayed in a LMS (see Fig. 1).

3.3 The importance of plan stability

The ultimate objective in planning is to construct plans for execution. However, when a plan is executed in a real environment it can encounter differences between the expected and actual context of execution. These differences can manifest as divergences between the expected and observed states of the world, or as a change in the goals to be achieved by the plan. In both cases, the original plan must be replaced with a new one. In replacing the plan an important consideration is *plan stability*. As proposed in [4], we use this term to refer to a measure of the difference a process induces between an original (source) plan and a new (target) plan. In general, we will be considering cases where the new plan is intended to solve a different, although related, problem to the one solved by the original plan. This means that there will inevitably be a difference between the plans. In the e-learning context it is extremely important to preserve as much as possible the LOs planned for each single student; in fact, it could be extremely disappointing if a completely new sequence of LOs is proposed to the students given a change in the current state, and it should be avoided as much as possible, especially if the original LOs can still be used.

Then, we decided to use a simple but very effective notion of plan stability [4, 12] based on the distance in terms of number of different LOs between two learning plans. Following the formalization proposed in [4], the *distance* between two plans is simply defined as the number of actions (LOs in our context) that appear in the first plan and not in the second, plus the number of actions that appears in the second plan and not in the first one. Given an initial learning plan that is no longer valid due to a change of the current state or to a change of the

domain representation, the notion of *plan stability* is simply defined in terms of the *distance* of the new solution plan w.r.t. the original plan.

4 Executing, Monitoring and Adapting E-learning Routes

The use of LMSs is important, mainly for the students but also for the teachers. The LMS identifies the instructional design per student, i.e. his/her learning route, can be visualised under the IMS-CP or SCORM specifications following the compilation criteria described in section 3.2. The LMS is therefore useful not only for navigation matters, but also to automatically monitor the student's progress and detect significant discrepancies between the current situation and the scheduled (expected) situation in a kind of *checkpoint* (see Fig. 2). These discrepancies appear due to changes on the background/profile information, the temporal constraints, the resource availability, or the execution of the LOs in themselves. Some examples of this are:

- The student's background is externally changed. For instance, the student is involved in an external language course, or has worked with many LOs in that language, and consequently (s)he has become more proficient in such a language. These improvements in the student's skills will allow him/her to choose now from a higher number of LOs.
- The learning style orientation of the student changes throughout the course execution, which entails a revision of the remaining LOs of the course. Some of them will remain valid, but others should be replaced to fit the new student's profile.
- The student has extra temporal constraints (getting a new job or being sick), and now (s)he has less time to accomplish the goals of the course, thus being unable to perform some LOs. This is likely to create an inconsistency when using *tightly-agenda* LOs.
- There is a change in the availability of the equipment, which is temporary unavailable. Or perhaps the student has now a better-equipped computer. In both cases, the learning route may require an adaptation process.
- During the course execution the student might fail a test or questionnaire, that is a checkpoint LO used to evaluate his/her comprehension on the course objectives. If this comprehension shows a low score, the student's performance will not be enough to attain the learning outcomes.

As can be noted, there are both *positive* and *negative* discrepancies. Positive discrepancies, such as having more available resources or when students' abilities are improved, do not invalidate the learning route, but they could lead to a better quality route, i.e. shorter makespan or higher reward plans. On the contrary, negative discrepancies make the learning route no longer executable, e.g. some resources are unavailable or the student fails an evaluation activity.

The changes in students' background, learning styles and temporal constraints must be modified directly by the students using the LMS interface. Changes related to the resource availability are usually updated by teachers,

and scores of evaluation activities can be input by the teachers or automatically calculated by the LMSs. With all these changes, a new (planning) problem with the same learning goals —although they can be also changed if desired— and a new initial state is created. After this, our way of proceeding is depicted in Fig. 2. When changes in the student’s profile are detected, we simulate the execution of the remaining part of the learning route, starting from the new state, in order to identify if it contains flaws, i.e. whether the prerequisites of LOs and the goals are satisfied or not (this validation process can be computed efficiently in polynomial time w.r.t. the number of actions in the remaining part of the plan [6]). 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 planner that will fix the flaw, that is automatic adaptation. 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 resources involved, and does not require any kind of validation by the teacher since the LOs have not changed. 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, once a new plan is computed by our system it must be always validated by the teacher before its execution, and the plan stability, in terms of number of actions of the original plan, 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 stored in the case base (plan library), if not already present, closing in this way the learning cycle as shown in Fig. 1.

5 Experimental Results

In this section we test the effectiveness of our adaptation approach *vs.* plan generation techniques when discrepancies appear while executing the learning routes. Particularly, we focus on: i) the CPU time required to repair (adapt or replan) the routes, ii) the best qualities in terms of higher reward plans, and iii) the best stability that can be obtained in a given deadline. We use 2 real, different Moodle courses (planning domains), on Discrete Maths and Natural Sciences, which are medium- and large-size, respectively. For each of the 2 courses, we have created 4 initial configurations (with 20, 40, 60 and 80 different students, respectively), and defined 10 variants per configuration, thus considering 88 planning problems in total (the 80 variants plus the 8 initial configurations). Each variant artificially simulates the changes that may occur during the route execution in an incremental way. That is, in the first variant some equipment is no longer available. The second variant maintains these changes and includes restrictions on the students’ availability; and so on for the other variants.

In addition to our adaptation approach, implemented on LPG-ADAPT [4], we have used two state of the art planners, SGPLAN6 and LPG¹. Since LPG and

¹ For a further description of these planners see <http://ipc.icaps-conference.org>

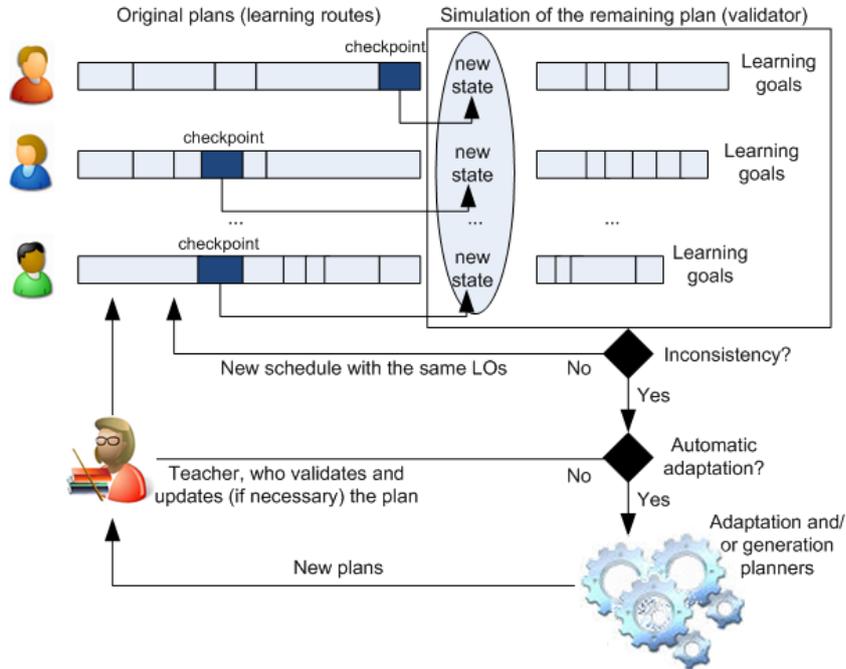


Fig. 2. Schema for monitoring and adapting the e-learning route.

LPG-ADAPT are stochastic incremental planners and different executions usually differ, we have performed 5 runs for each planning problem and taken the median of these values for our plots. All tests were performed on an Intel(R) Xeon(TM), CPU 2.40GHz, 2GB of RAM, and censored after 10 minutes. In our tests, the input plan (i.e. the learning route) to be adapted by LPG-ADAPT was obtained by using the best quality plan generated by LPG and SGPLAN6 on the initial-configuration planning problem used to create the corresponding variants.

Fig. 3 depicts the results: the time taken to produce a solution—the first one for LPG and LPG-ADAPT— (top); the quality of the generated routes (middle); and the stability, in terms of distance of the new routes to the original ones (bottom). We show the best distance and plan quality across all plans produced in the entire optimisation phase². The results demonstrate that plan adaptation is at least as fast as replanning, and usually faster. Obviously, adaptation shows less useful when the changes are significant and fixing the route requires more effort than simply discarding it and rebuilding a new one from scratch. But the benefits for investing this effort can be seen in terms of stability. On the other hand, the adaptation sometimes comes at a price in terms of quality, as the

² Note that the first plan generated by LPG and LPG-ADAPT, the best quality plan and the best distance plan could be different plans. It depends on the teacher's preferences to give more importance to the plan quality or to the plan stability by selecting the most appropriate solution plan during the validation process.

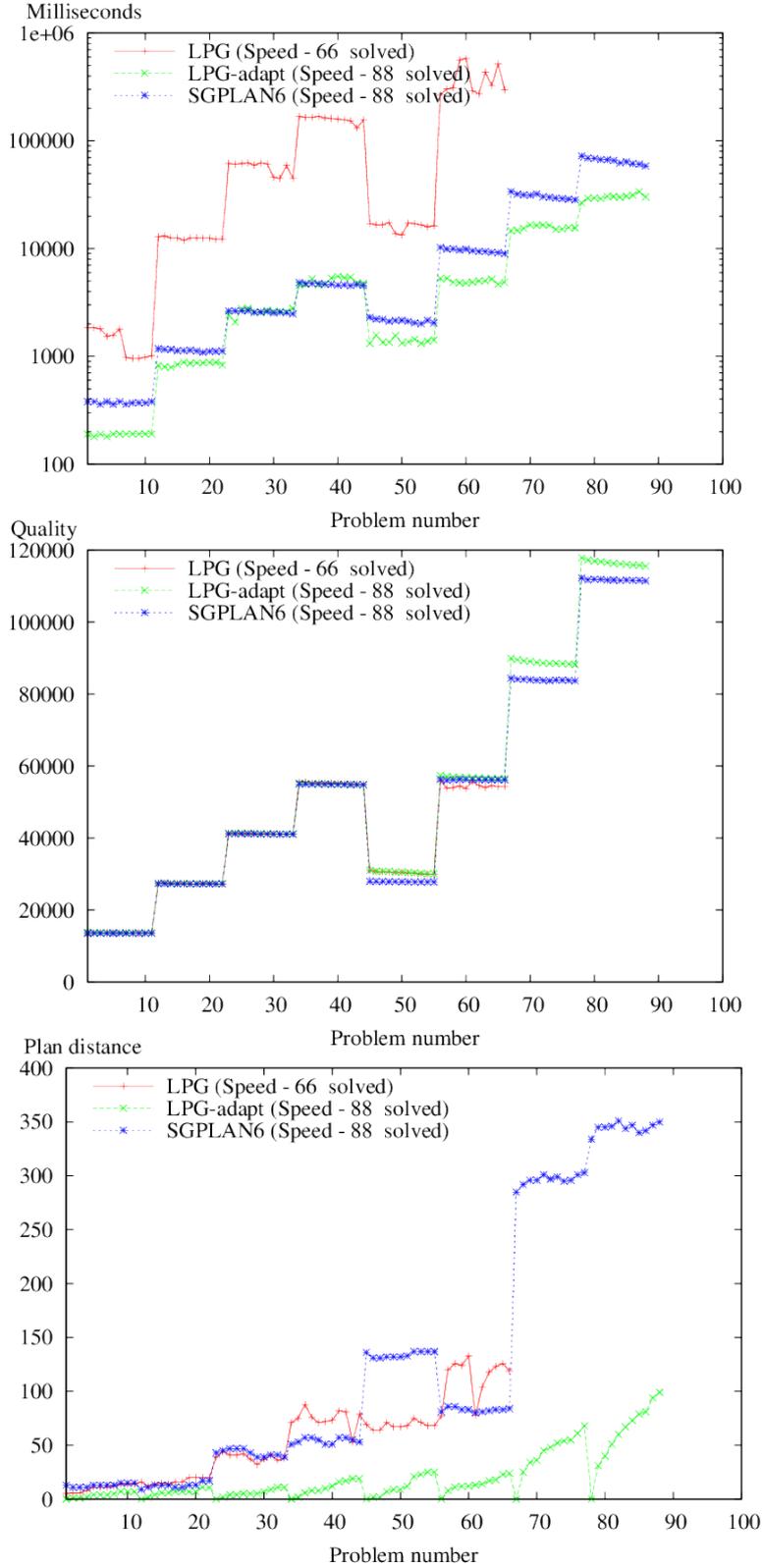


Fig. 3. CPU time (on a logarithmic scale), plan qualities and number of different LOs w.r.t. the input plan of the adaptation —small values are preferable except for quality. We compare our adaptation approach (LPG-ADAPT) vs. replanning (LPG and SGPLAN6).

route is adapted to fit a new configuration rather than constructed expressly for it. But our experiments show that the quality for adaptation can be better than for replanning, particularly in the most complex problems (see Fig. 3). We therefore compare the best relative qualities of plans generated by adaptation and replanning within a certain CPU time limit. Finally, the best values for stability are outstandingly achieved in plan adaptation. While replanning generates routes that are consistently very different to the original ones (see SGPLAN6 for a clear example), the differences between the adapted plan and the original plan are very small. This indicator is very appealing in an e-learning setting as the students/teachers do not want to deal with an entirely new learning route after a little change happens during execution. Quite the contrary, students and teachers prefer a kind of *inertia* in the learning routes that facilitates the learning process.

6 Conclusions

Personalisation of e-learning routes is essential for both educational and enterprise organizations, as it supports a continuous and fruitful lifelong learning process. In this paper, we have presented a flexible way that consists in the compilation of the course+students characteristics into a a planning problem to find these routes. But once a learning route is generated, how the students execute such a route (and use its LOs) is also a challenging aspect. Monitoring the route may detect inconsistencies that can turn it invalid, and an adaptation process becomes crucial.

We have proposed an approach for executing and monitoring learning routes that uses an adaptation method to repair unexpected discrepancies (and to improve the quality of the original plan when possible). This approach has some advantages, which are the main contributions of this paper: i) it is implemented on top of a standard LMS platform (Moodle), and all the information retrieved and produced is mapped from, and to, e-learning standards; ii) the adaptation technology considers, not only students' preferences on the course, but also teachers'; iii) it allows dynamic changes both on the students' profiles (planning problem) and on the course structure (planning domain); iv) it provides multi-optimisation methods, useful for modern planners and for navigation in LMSs.

As part of our current work, we are addressing two issues. First, to extend the notion of plan stability to deal with extra temporal and resource constraints (e.g. the use of a laboratory at a specific time or the participation of the same set of students in group activities). The idea is to include structural properties of the original plan expressed as *preferences* to be maintained in the new plan. Second, to implement a collection of Web services to be used as a standard interface for LMS-based agents to monitor changes in the student's profile, course composition and description. This will allow us to evaluate our approach in a higher number of real students and situations.

Acknowledgments. 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. Abdullah, N., Davis, H.: Is simple sequencing simple adaptive hypermedia? In: Proc. ACM Conference on Hypertext and Hypermedia. pp. 172–173 (2003)
2. Camacho, D., Pulido, E., Rodriguez-Moreno, M., Carro, R., Ortigosa, A., Bravo, J.: Automatic course redesign: Global vs. individual adaptation. *Journal of Engineering Education* 25(6), 1270–1283 (2009)
3. Castillo, L., Morales, L., Gonzalez-Ferrer, A., Fdez-Olivares, J., Borrajo, D., Onaindia, E.: Automatic generation of temporal planning domains for e-learning. *Journal of Scheduling* 13(4), 347–362 (2010)
4. Fox, M., Gerevini, A., Long, D., Serina, I.: Plan stability: Replanning versus plan repair. In: Proc. 16th Int. Conference on Automated Planning and Scheduling (ICAPS-2006). pp. 212–221. AAAI Press (2006)
5. Garrido, A., Onaindia, E., Morales, L., Castillo, L., S., F., Borrajo, D.: Modeling e-learning activities in automated planning. In: Proceedings of the 3rd International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS-ICAPS 2009). pp. 18–27 (2009)
6. Howey, R., Long, D., Fox, M.: Validating plans with exogenous events. In: Proc. 23rd UK Planning and Scheduling SIG Workshop. pp. 78–87 (2004)
7. Idris, N., Yusof, N., Saad, P.: Adaptive course sequencing for personalization of learning path using neural network. *Int. J. Advance. Soft Comput. Appl.* 1(1), 49–61 (2009)
8. Kontopoulos, E., Vrakas, D., Kokkoras, F., Bassiliades, N., Vlahavas, I.: An ontology-based planning system for e-course generation. *Expert Systems with Applications* 35(1-2), 398–406 (2008)
9. 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
10. Peachy, D., McCalla, G.: Using planning techniques in intelligent systems. *International Journal of Man-Machine Studies* 24, 77–98 (1986)
11. Perez-Rodriguez, R., Rodríguez, M., Anido-Rifón, L., Llamas-Nistal, M.: Execution model and authoring middleware enabling dynamic adaptation in educational scenarios scripted with PoEML. *Journal of Universal Computing Science* 16(19), 2821–2840 (2010)
12. Srivastava, B., Nguyen, T., Gerevini, A., Kambhampati, S., Do, M., Serina, I.: Domain independent approaches for finding diverse plans. In: Proc. Int. Joint Conference on AI (IJCAI-2007). pp. 2016–2022 (2007)
13. Ullrich, C., Lu, T., Melis, E.: Just-in-time adaptivity through dynamic items. In: User Modeling, Adaptation, and Personalization Seventeenth International Conference, UMAP 2009. vol. LNCS 5535, pp. 373–378. Springer-Verlag (2009)
14. Ullrich, C., Melis, E.: Pedagogically founded courseware generation based on HTN-planning. *Expert Systems with Applications* 36(5), 9319–9332 (2009)