

A SYSTEM FOR GENERATING PEDAGOGICAL SCENARIOS FOR SERIOUS GAMES

Aarij Mahmood Hussaan

Université de Lyon, CNRS, Université Lyon 1, LIRIS, UMR5205, F-69622, Lyon, France

Karim Sehaba

Université de Lyon, CNRS, Université Lyon 2, LIRIS, UMR5205, F-69676, Lyon, France

Keywords: Adaptive scenario, Serious games, Users interaction traces, Users profile, Handicap.

Abstract: Serious games are often used in education as they provide an excellent opportunity for learning. This paper presents our proposition of a system capable of generating adaptive learning scenarios for serious games. Thus, we have presented the architecture of this system. The domain knowledge is organized in three layers. These layers include the domain concepts, the pedagogical resources and the serious game resources. The approach we propose is capable to keep in account the specificities of serious games, the user profile and his/her goals while generating scenarios. Furthermore, it also uses the interaction traces as knowledge sources in the adaptation process. To test the applicability of our system we are working on the Project CLES (Cognitive and Linguistic Element Stimulation). This project targets producing serious games for users with cognitive disabilities. We've presented an example of how our system will generate a pedagogical scenario.

1 INTRODUCTION

Serious games are defined as *"a mental contest, played with a computer in accordance with specific rules, which uses entertainment to further government or corporate training, education, health, public policy, and strategic communication objectives"* (Zyda, 2005). Hence, the idea to learn while playing games is very attractive for most of the users.

In the context of systems capable of generating pedagogical courses and serious games. The objective of this study is to propose a system that can generate dynamically **adaptive pedagogical scenarios** keeping into account the following properties:

- The ability to be utilized in a variety of serious games taking into account their specificities.
- Use of users' interaction traces as knowledge sources in the adaptation process.
- The ability to be presented to cognitively handicapped users.

Indeed, a serious game like any other game has many elements namely game characters, narrative, game world, game world objects, goals, rules, pedagogical elements, etc. A review about the design of

different serious games can be found in (Dondlinger, 2007) (Michael, David R. And Chen, 2005). To create pedagogical scenarios that can be utilized in many serious games, it is necessary to keep in account various aspects of serious game design that are common in a vast amount of games. This will help in keeping the scenarios usable with different games. Furthermore, it is also necessary to make sure that the games can be easily playable for handicapped users.

Another requirement is the use of modeled users' interaction traces as knowledge sources for the adaptation process. A user's interaction traces can be defined as *"a history of learner's actions collected, in real time, from his/her interaction with a computer system"* (Sehaba et al., 2009). So these traces will also help the system in keeping an account of the history of user's actions i.e. the evolution of the user's competence can be detected and be used by the system. Furthermore, their modeling will help us in detecting the state in which the user finds himself in while playing the game, enabling the system to provide help (hints, etc) to the user in case of need. Furthermore, the level of the game can be adjusted according to the user's performance. Furthermore, the system can also make sure that the same exercises are

not repeated many times to a user.

The problem of generating pedagogical scenarios for a user is not new and is addressed previously by many authors like (Brusilovsky, 1993) (Ullrich and Melis, 2009) (Specht et al., 2001) (Karampiperis and Sampson, 2005). These works focus only on generating *pedagogical scenarios*. Therefore, they do not take into account the specificities of serious games. Consequently, they cannot be used easily with them. Furthermore, none of these systems neither model the users' interaction traces nor use them as knowledge sources in their adaptation process.

Similarly, the idea of using serious games in education is not new and is being employed by many systems like (Morenoger et al., 2007) (Torrente et al., 2009) (Carron, Thibault And Marty, Jean-Charles And Heraud, 2008). However, most of the time these systems lack the notion of a pedagogical scenario. Furthermore, almost all of the systems neither model users' interaction traces nor use them for adaptation. (Carron, Thibault And Marty, Jean-Charles And Heraud, 2008) do use the interaction traces but their system neither is usable with different games nor their scenarios are adaptive.

The remainder of this article is organized as follows, the next section will present the application's context for our work. In section 3 the architecture of our system will be presented. This is followed by the presentation of our course generator in section 4. In section 5 we present an example of scenario generation of our system. This article is then concluded by presenting a brief conclusion in section 6.

2 APPLICATION'S CONTEXT

This work concerns the project CLES (Cognitive and Linguistic Element Stimulation). CLES is a project funded by the French Ministry. It is conducted in partnership with the laboratories d'Étude des Mécanismes Cognitifs (EMC)¹, Laboratoire des Usages en Techniques d'Information Numériques (LUTIN)² and the society GERIP³. This project will create a "Serious Game" based environment with integrated tools based on educational game for the reeducation and the cognitive simulation of children, adults and seniors. This project will cover all the themes of speech (spoken language and written language), memory, calculation disorders, attention disorders, cognitive impairments, assessment of the cognitive skills, voice disorders and

¹<http://recherche.univ-lyon2.fr/emc/rubrique-2-Presentation.html>

²http://www.lutin-userlab.fr/_pages/english/

³<http://www.gerip.com/about/presentation.html>



Figure 1: Configuration screen of "Long-Vue".

audio and visual perception disorders. Therefore, for every pathology, this project aims to develop a game that is focused on the specific deficiencies while maximizing, through techniques employed in video games, their cognitive ergonomics..

Based on these components, our contribution, in this project, consists in developing a module capable of generating personalized courses according to the challenges to be faced and progress of each patient. This module should therefore keep in account

- The specificities of the serious game
- What the practitioner has prescribed his patients
- The knowledge base of the treatments available for the pathology
- Results of the previous exercises of the user

This module will also be able to help the practitioner in monitoring the progress of his patients in each learning session. The latter, once designed, will be subject to a validation process including "clinical tests" that will be conducted with project partners.

The games that have been already developed in this area contains games testing different cognitive abilities of a person. The figure 1 presents the screen shot of a game called *Long – Vue*. This game tests the working memory capacity of its users. It asks the user to recall a word or image after showing a list of words or images. The parameters of the game let the user to adjust the length of the list of words or images, the time a single word is displayed on the screen, font sizes, size of characters in the words of the list, etc.

3 SYSTEM ARCHITECTURE

Figure 2 shows the general architecture of the system. The different components of the system are as follows : domain model, resource model, game model, presentation model, user profile and course generator. The domain model consists of the concepts related to the pedagogical domain (the domain to be

taught by the system) and the concepts related to the physical and cognitive capacities of the user. The domain concepts are related to each other with one or more relations. These relations are didactic in nature. We've identified some relations that are used in the system. These relations are: Has-Parts, Required, Order, Type-Of and Parallel. Some of these relations are used by other authors as well (Karampiperis and Sampson, 2005) (Duitama et al., 2005).

Has Parts: $HP = (x, y_1, y_2, y_3 \dots y_n)$: the concept x is composed of the sub-concepts $y_1, y_2, y_3 \dots y_n$. To learn x it is necessary to learn all the concepts $y_1, y_2, y_3 \dots y_n$.

Requires: $R = (x, y)$: to learn x it is necessary to have sufficient knowledge of y .

Order: $O = (x, y)$: it is preferable to present concept x before y

Type-Of: $TO = (x, y)$: concept y is a type of concept x . This relation can be considered as a Specialization relation.

Parallel: $P = (x, y)$: the concept x and y are parallel i.e. if one is selected then the other should also be selected during a course generation

The resource model contains all the *pedagogical resources* for e.g. definitions, examples, tests etc. These resources are related to one or more domain concepts. The game model contains all the *resources of serious games* for e.g. bed, lamps, paintings, non-player characters (NPC), drawers etc. These resources *uses* the pedagogical resources to be presented to the user.

The three kinds of knowledge (pedagogical domain, pedagogical resources and serious game resources) are kept separate in the system. This separation is necessary to make reduce dependency of one type of knowledge on the other. This helps in making the system more generic i.e. usable for different domains. Furthermore, this also helps in making the system usable with a variety of serious games.

The purpose of the presentation model is to organize the pedagogical resources presented to the user. In fact, this model contains two sub models namely 'scenario model' and 'test model'. The scenario model defines the structure of the scenario for e.g. a scenario starts by presenting two definitions followed by an example and an exercise. The test model describes the system's behavior on *test* type resources for e.g. an easier *test* is presented after each failure. The selection of these models can either be made by the user or by the teacher (expert) for the student. The structure of the scenario model can be of the form as defined in (Ullrich and Melis, 2009).

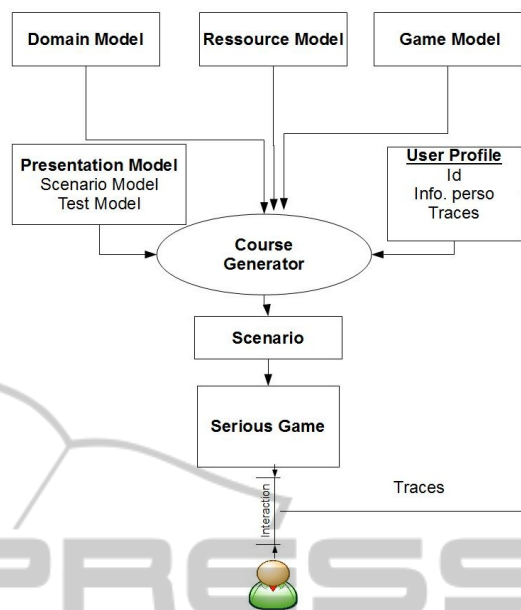


Figure 2: General architecture of our system.

The user profile contains the user's information. This information includes the user's personal details and the progress made by the user. The traces of his interaction with the system is also kept in the profile. These traces help in keeping an account the concepts the user has already visited and user's mastery of those concepts. The pedagogical and serious game resources s/he has already seen. The order in which the user has visited the resources. The evaluation of his progress. Traces help in making sure that the same resources are not repeated unnecessarily to the user. This repetition can be discouraging for the user.

The course generator is where all the generation of courses take place. The working of this module is discussed in detail in the following section.

4 COURSE GENERATOR

This is the heart of our system. The main purpose of the *Course Generator (CG)* is to generate a scenario according to the profile of the user and his objectives. Furthermore, during the interaction this module is capable of adapting the scenario according to the users' traces. The process of this module is shown in the figure 3. In this article we don't give the details of the implementation of the algorithms used by the course generator. We give here only the general principle of its functioning.

In the 1st step the learning goals (selected by the users or by the system according to the user's profile) in the form of target concepts are provide to

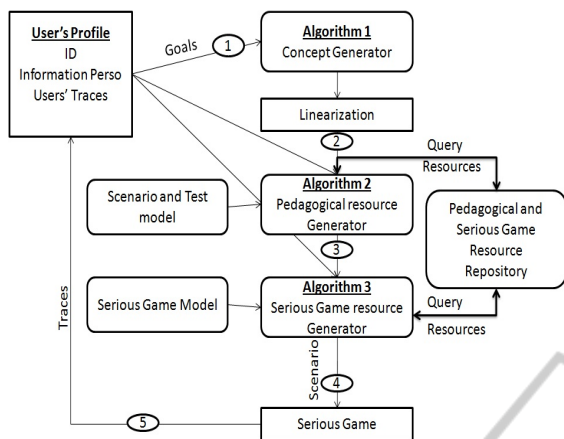


Figure 3: Course Generation process.

the course generator. Based on these learning goals and the user's profile a list of *result concepts* are selected. These *result concepts* are necessary to teach the learning goals. This generation is done by *Algorithm 1*. The working of this algorithm goes as follows: for every concept in the learning goals the algorithm consults the user's profile to verify whether or not the concept is sufficiently known by the user. If it is known by the user this concept is ignored. If the concept is not known then the algorithm searches for related concepts (recall that a concept can be related to other concepts via relations defined earlier). If there are related concepts that are not known by the user then these concepts are added to the list of *result concepts*. The algorithm is called recursively for every concept in the *result concepts* list. The working is also described in figure 4. After the generation of the concepts, to be taught, a linearization phase is applied to them. This linearization is described in (Vassileva, 1995) and it is utilized by (Capuano et al., 2002) among other authors. The objective of this phase is to order the generated concepts according to the Order relation. Furthermore, the algorithm can generate a concept multiple times; this phase will help to eliminate this anomaly.

In the 2nd step, these concepts are sent as input to another algorithm *Algorithm 2*. The purpose of this algorithm is to generate pedagogical resources based on the generated concepts (1st step), the scenario and the test model selected by the user. This is done by the algorithm by querying the "pedagogical and serious game resource repository" keeping into the account the user's profile and the selected models. In return the repository returns to the algorithm the queried resources.

In the 3rd step the selected resources are given to the algorithm *Algorithm 3*. The purpose of this algorithm is to associate the selected pedagogical

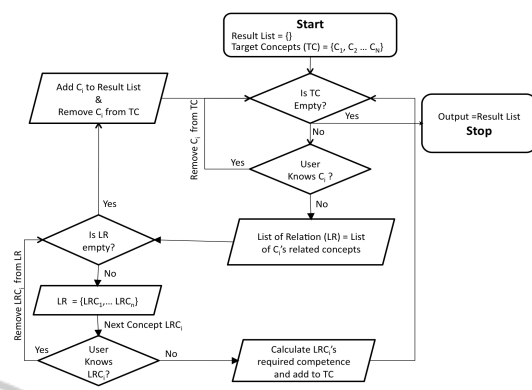


Figure 4: Working of the *Algorithm 1*.

resources with appropriate serious game resources. This is done by selecting an appropriate serious game model and then querying the resource repository for appropriate resources keeping into account the user's profile. The result of this query is a set of serious game resources initialized by pedagogical resources. In the 4th, these resources are presented to the application in the form of an scenario.

The user starts to interact with the serious game and as a result the user's interaction traces are generated. In the 5th step, these traces are transferred to the user profile. In the user profile, these traces are used as knowledge sources to update the profile and modify the scenario if necessary.

5 ILLUSTRATION

The application's context (Project CLES) has been mentioned and described earlier in section 2 . Here we'll give an example of the scenario generated by our system for this application.

Project CLES has created a serious game based environment. This game is an adventure game. The main protagonist of this story is a treasure hunter called Tom O'Connor. His objective is to find a hidden relic. The story places Tom in one of several rooms. Each room is connected to other rooms. Each room represents one of the cognitive domains (described above). Each room has multiple game objects like (bed, lamps, paintings etc). Hidden behind some of these objects are challenges. These challenges are in the form of games. A screen shot of one of these games is shown in figure 1. The user is required to click one of these objects to access the challenges hidden behind them. Then the user is required to solve these challenges in order to progress to other rooms. A screenshot of the CLES's environment is shown in figure 5.

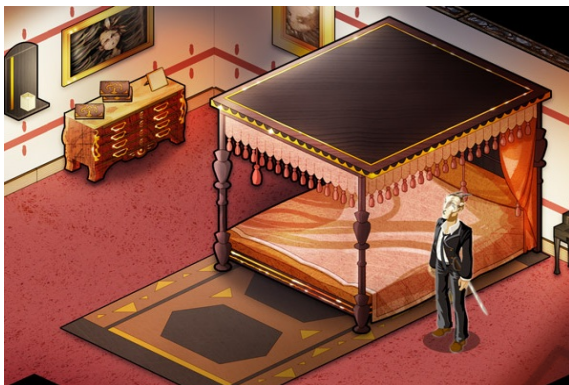


Figure 5: Screenshot of the game.

Our contribution in this project is the generation of pedagogical scenarios given the pedagogical objectives and the profile of the user. This domain has eight main concepts and then these main concepts are further divided. These main concepts are 1) perception, 2) attention, 3) memory, 4) visual-spatial 5) logical reasoning 6) oral language 7) written language, 8) transversal competencies. The sub structure of all of these eight domains is quite large. Hence, for the sake of space we'll show the modeling of only one of the structures in the figure 6.

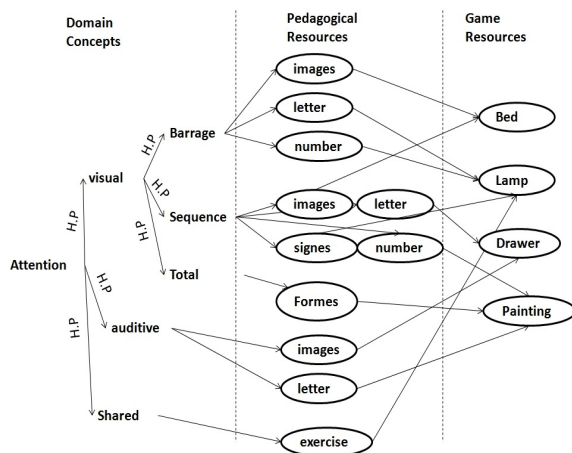


Figure 6: Domain Model of the concept Attention'. The arrows labeled H.P. are Has-Parts relations.

Each and every atomic concept is associated with a number of pedagogical resources. In this case these resources are only of the type 'test'. These tests vary in difficulty. For e.g. an element is hidden in a complex image and it is required from the user to find that element as quickly as possible, many pair of glasses are hidden in a decorated room and the user is required to find these glasses, etc. Furthermore, selected pedagogical resources are used by selected serious game resources to be displayed to the user.

The scenario model defines that each concept should be treated separately i.e. the user is presented one concept at a time. Initially user is assigned with one of the predefined profile (Stereotype) based on his age. This profile will be evolved according to the user's over time. Furthermore, the profile is also used in the system to provide the adaptation.

Now, let us suppose that the *target concept* to be learned by the system is *attention visual*. Also suppose, that the knowledge of the user about the conception *attention visual* is declared 40% in the profile. Furthermore, the level of knowledge the user wants to acquire is 50%. The user selects the appropriate presentation models (scenario and test model) and serious game model. These models along with the target concept are given as input to the *course generator*. Firstly, the CG, using the *Algorithm 1* will generate the list of concepts require to teach the goal concept. For the target concept *attention visual* the selected concepts are: 1.Barrage → 2.Sequence → 3.Total.

Now let us suppose that the selected scenario model have the following structure: 1.Test → 2.Test → 3.Test.

For the concept *Barrage*, based on this scenario model and the user's profile the *Algorithm 2* will give the following resources :

- 'Images' of *Barrage* (difficulty 50%)
- 'letter' of *Barrage* (difficulty 50%)
- 'number' of *Barrage* (difficulty 50%)

Supposing that we have a repository of serious game containing the resources like *Chair, Table, Statue*. Based on the game model and the user's profile the *Algorithm 3* will give the following serious game resources :

- 'Images' is hidden behind *Bed*
- 'letter' is hidden behind *Painting*
- 'number' is hidden behind *Drawer*

These resources are presented to the user via the serious game. The user interacts with the serious game and consequently traces are generated. These traces are stored in the user's profile. Furthermore, the profile is updated based on these traces and the scenario is modified (if necessary). For example, if the user fails the 1st test another test with a difficulty of 40% is presented to the user. This exercise will be selected dynamically while the user is interacting with the 1st exercise. This selection is done based on the *test model*.

6 CONCLUSIONS

In this paper, we presented a system that is able to generate pedagogical scenarios while keeping into account the specificities of serious games and users' interaction traces. We also presented the modeling of this system along with a worked example of how the system will generate a scenario.

There are systems that are designed to generate pedagogical scenarios. However, these systems only focus on pedagogy and do not take into account the specificities of serious games. Hence, they can't be used with serious games. Furthermore, they do not make use of modeled users' interaction traces in their adaptation process. They are limitations of these systems from our point of view.

Similarly, there are some serious game systems that are used to provide education. Some of these systems have a notion of pedagogical scenario but these scenarios are static in nature and are not modified according to the user's profile. Like course generators these systems also do not take into account the users' interaction traces for adaptation purposes.

The system we proposed in this paper addresses these problems. The solution is presented in the form of a system. The architecture of this system is presented and detailed. This profile includes the information about the user, his competence and the traces of his/her interaction with the serious game. These traces are used as knowledge sources in the adaptation process. The domain knowledge is divided into three layers. These layers include the domain concepts, the pedagogical resources and the resources of serious games. Furthermore, the working of the *Course Generator* is also presented. This course generator is responsible for the generation of the serious game scenario.

The context of our work's application is presented in this paper. We are creating a prototype of the system that implements this architecture. Then this system will be tested on a real time system with real students. These students will have a cognitive impair. These tests will present us with an opportunity to validate our approach. After the validation of this approach we'll test our system in other domains to validate the general applicability of our system.

REFERENCES

- Brusilovsky, P. (1993). Task sequencing in an intelligent learning environment for calculus. In *Seventh International PEG Conference*, pages 57–62.
- Capuano, N., Gaeta, M., Micarelli, A., and Sangineto, E. (2002). An integrated architecture for automatic course generation. In *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT 02)*, number Section 4, pages 322–326. Citeseer.
- Carron, Thibault And Marty, Jean-Charles And Heraud, J.-M. (2008). Teaching with Game Based Learning Management Systems : Exploring and observing a pedagogical. *Simulation & Gaming*, 39(3):353—378.
- Dondlinger, M. (2007). Educational video game design: A review of the literature. *Journal of Applied Educational Technology*, 4(1):21–31.
- Duitama, F., Defude, B., Bouzeghoub, A., and Lecocq, C. (2005). A framework for the generation of adaptive courses based on semantic metadata. *Multimedia Tools and Applications*, 25(3):377–390.
- Karampiperis, P. and Sampson, D. (2005). Adaptive learning resources sequencing in educational hypermedia systems. *Educational Technology & Society*, 8(4):128–147.
- Michael, David R. And Chen, S. L. (2005). *Serious Games: Games that educate, train, and inform*. Muska & Lipman/Premier-Trade.
- Morenoger, P., Sierra, J., Martinezortiz, I., and Fernandezmanjon, B. (2007). A documental approach to adventure game development. *Science of Computer Programming*, 67(1):3–31.
- Sehaba, K., Encelle, B., and Mille, A. (2009). Adaptive TEL based on Interaction Traces. In *In AIED 09 (14 International Conference on Artificial Intelligence in Education) workshop on "Towards User Modeling and Adaptive Systems for All (TUMAS-A 2009): Modeling and Evaluation of Accessible Intelligent Learning Systems"*.
- Specht, M., Kravcik, M., Pesin, L., and Klemke, R. (2001). Authoring adaptive educational hypermedia in WINDS. *Proceedings of ABIS2001, Dortmund, Germany*, 3(3):1–8.
- Torrente, J., Moreno-Ger, P., Fernández-Manjón, B., and del Blanco, A. (2009). Game-like Simulations for Online Adaptive Learning: A Case Study. In *Proceedings of the 4th International Conference on E-Learning and Games: Learning by Playing. Game-based Education System Design and Development*, page 173. Springer.
- Ullrich, C. and Melis, E. (2009). Pedagogically founded courseware generation based on HTN-planning. *Expert Systems with Applications*, 36(5):9319–9332.
- Vassileva, J. (1995). Dynamic courseware generation: at the cross point of CAL, ITS and authoring. In *Proceedings of ICCE*, volume 95, pages 290–297.
- Zyda, M. (2005). From visual simulation to virtual reality to games. *Computer*, 38(9):25–32.