Dynamic Scenario Adaptation Balancing Control, Coherence and Emergence

Camille Barot, Domitile Lourdeaux and Dominique Lenne
Heudiasyc - UMR CNRS 7253, Université de Technologie de Compiègne, 60200 Compiègne, France

Keywords: Planning, Virtual Environments for Training.

Abstract: As the industrial world grows more complex, virtual environments have proven to be interesting tools to train workers to procedures and work situations. To ensure learning and motivation, a pedagogical control of these environments is needed. However, existing systems either do not provide control over running simulations, limit user agency, need the authors to specify *a priori* all possible scenarios, or allow for incoherent behaviours from the simulated technical system or the virtual characters. We propose in this paper a model for a dynamic and indirect control of the events of a virtual environment. Our model aims to ensure the control, coherence, and emergence of situations, in virtual environments designed for training in highly complex work situations.

1 INTRODUCTION

With the growing complexity of work situations and procedures, training has become a key issue in the industrial world. In the past decade, there has been an increasing interest in the use of virtual reality environments for training. Realistic simulation of work situations allows for an efficient training, especially when cost, accessibility, or dangerosity aspects prevent learners from being put in genuine work situations. However, the framed simulations that are used to train workers to technical gestures and standard procedures are no longer sufficient to answer all training needs, especially when stepping out of initial training to address continuous training of experienced workers, or when critical domains such as risk management are concerned. In such cases, the virtual environment scenario has to be controlled dynamically according to pedagogical rules, in order to provide interesting and relevant situations, adapted to the learner's profile and needs. Yet, controlling such complex environments is far from being trivial, and existing systems often have to make trade-offs on either the control, coherence (i.e. perceived consistency in characters motivations and technical system reactions), or emergence of new situations. We believe that these three aspects need to be respected so that the virtual environments can be used for real training sessions. Therefore, we propose a scenario adaptation process that allows our system to control the unfolding of events in highly complex simulations containing virtual autonomous characters, while keeping the global scenario and individual behaviours coherent. Moreover, this process is dynamic, so that the adaptation can happen during the course of the simulation, to cope with learners agency and fit their evolving training needs. After presenting the limits of existing systems, we will expose our model for a dynamic scenario adaptation. We will illustrate our proposal with a scenario example, then discuss its limitations. We will then conclude and expose the perspectives we foresee for this work.

2 RELATED WORK

Most virtual environments designed for training or educational purposes rely on a pedagogical scenario, which is a sequence of learning activities the user has to perform, whether a predetermined sequence of scenes (Magerko et al., 2005), or a prescribed procedure the user has to execute (Mollet and Arnaldi, 2006). Yet, in order to stay within the boundaries of these predefined paths, these virtual environments offer a strong guidance to the trainee, often stopping them whenever they make a mistake or an action that does not belong to the training scenario. By limiting the user's freedom of action, these systems prevent trial-and-error approaches. Moreover, scenarios definition requires a large amount of work when the training addresses long procedures or complex situations.

Barot C., Lourdeaux D. and Lenne D..
Dynamic Scenario Adaptation Balancing Control, Coherence and Emergence.
DOI: 10.5220/0004213802320237
In *Proceedings of the 5th International Conference on Agents and Artificial Intelligence* (ICAART-2013), pages 232-237
ISBN: 978-989-8565-39-6

On the other hand, some environments focus on the simulation part by opting for the "sandbox" approach. They let the user act freely as the simulation evolves and reacts to their actions (Shawver, 1997). In these environments, the only pedagogical control is that of the initial state of the world. However, without any real-time pedagogical control, the efficiency of the training is not guaranteed. The simulation could go in any direction, yet we would want it to be relevant to the profile and current state of the trainee.

One approach for ensuring both user agency and pedagogical control is to define a multilinear graph of all possible scenarios. In (Delmas et al., 2007), the set of possible plots is thus explicitely modelized through a Petri Network. However, when the complexity of the situations scales up, it becomes difficult to predict all possible courses of actions. Especially when the training aims towards difficult coactivity situations, the decision making processes and emotions expressed by the virtual characters have to be believable, and therefore are often based on complex psychological models. In this case, it becomes impossible to foresee all possible combinations, and the virtual characters have to be given some autonomy in order for the scenarios to emerge from their actions.

To combine autonomous characters and a global scenario control is however fundamentally problematic: the controlling entity cannot influence autonomous characters behaviour unless they provide specific "hooks". And indeed, most of the environments that include complex, emotional characters, provide only semi-autonomous characters, like in *Scenario Adaptor* (Niehaus and Riedl, 2009). These characters can be given orders, whether at behavioural level, or on a higher, motivational level. The main weakness of this approach is that nothing ensures that the global behaviour of the characters will stay coherent. Yet coherence, especially in training environments, is essential to maintain to ensure the user's understanding of what is going on, as shown by (Si et al., 2010).

Few systems combine a global control of the simulation with the possibility of new situations to emerge from user actions or characters autonomous behaviour, all the while ensuring their coherence. An attempt to unite these different aspects has been made in *Thespian* (Si et al., 2009), by computing characters motivation at the start of the simulation so that the events would unfold according to an human-authored plot. However, this system doesn't allow dynamic scenario adaptation, in that we would like not to have a predefined plot but one that changes in real-time according to what learning situations are considered relevant in line with the user's activity.

3 PROPOSITION

3.1 Approach

As we aim to train to complex work situations with notable human-factors component, we adopt a character-based approach, using autonomous cognitive characters in order for such situations to emerge from both their interactions and those of the learner.

We propose a scenario adaptation module called SELDON (ScEnario and Learning situations adaptation through Dynamic OrchestratioN) that aims to ensure a pedagogical control over a complex simulation, without restraining the emergence of new situations or disturbing the coherence of objects or characters' behaviours. Our model lets the user act freely, and indirectly adjusts the unfolding of events. The scenario adaptation occurs not only at the start of the simulation but during its course, by dynamically generating learning situations that would be relevant to learner's profile and activity traces, then altering the scenario in real-time to guide him towards these situations.

SELDON is composed of two modules: TAILOR and DIRECTOR. TAILOR produces learning situations and constraints over the global scenario based on the current state of the learner (Carpentier et al., 2013). This paper focuses on DIRECTOR, which is in charge of generating a scenario respecting these constraints. The global scenario adaptation process is described in Figure 1, here shown whithin the HUMANS platform (Carpentier et al., 2013).

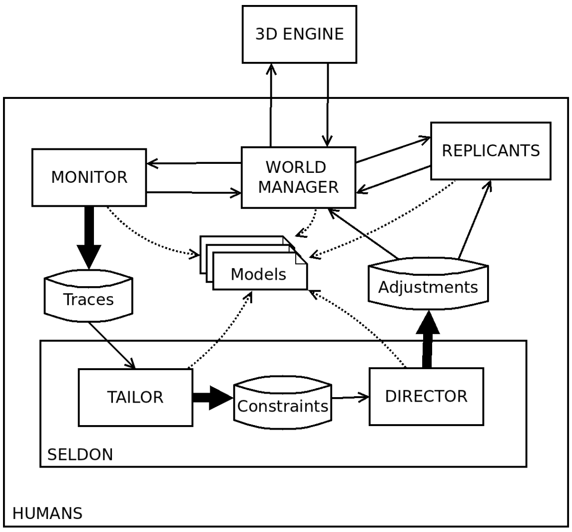

Figure 1: System architecture.

3.2 Scenario Adaptation Process

The evolution of objects states and the virtual characters actions in the HUMANS platform are deterministic and ruled by a domain model and an activity model, both described in (Carpentier et al., 2013). As DIRECTOR uses the same set of models, it can predict the unfolding of events (i.e. state changes) from a given initial situation (i.e. a set of states), and thus control the scenario by changing this initial situation.

However, such adjustments could only happen at the start of the simulation, while some of the states might not have to be initialized right away. For instance, the *broken* state of a *spring* object is crucial in determining the happening of a leak. However, until this leak happens, this state remains unknown to the learner, as it is not associated with a graphical representation. The *late commitment* (Swartjes, 2010) of such particular parameters can thus be used to direct the scenario dynamically. The system would then be able to adapt to changes in the pedagogical objectives, and to cope with the user's deviation from what had initially been planned.

The DIRECTOR module is in charge of producing a set of adjustments that would adapt the scenario according to the constraints that are given by TAILOR. This process is presented in Figure 2: first, DIRECTOR selects a set of partially ordered plot points; then, these plot points are instantiated and used as *landmarks* to plan a scenario graph consisting of both the actions and behaviours of the simulation's agents that are wanted in the scenario, and the adjustments needed for them to occur; finally, it tracks the changes in the environment in order to check that the events unfold according to the scenario, and triggers the adjustments when they are needed.

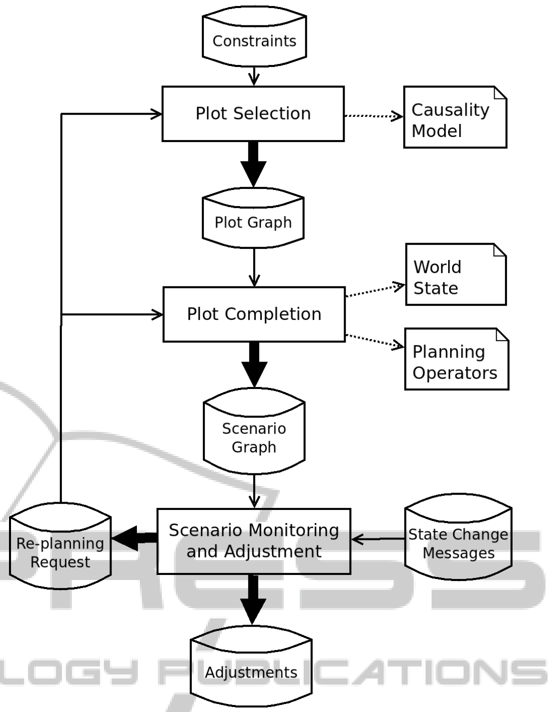

Figure 2: DIRECTOR scenario adjustment process.

3.2.1 Input: Constraints

DIRECTOR takes as input two types of constraints:

1. *Situation Constraints*: the situations that the user should encounter, or that should be avoided. One of these situations is tagged as the *goal situation*. Each constrained situation s is associated whith a *desirability* value $des \in [-1,1]$.
2. *Metric Constraints*: global constraints on the scenario, such as its complexity or believability (see Table 1). Each one is associated with an acceptable interval $I \subset \mathbb{R}$ and a strength $str \in [-1,1]$.

3.2.2 Output: Adjustments

To influence the simulation without modifying objects states or giving orders to the virtual characters, DIRECTOR can request three types of adjustments:

1. *Late Commitments*: several states are marked as *late commitable* during the domain definition, and their initial values can be specified during the simulation. Those states can be object states (e.g. amount of liquid in a container) or character states (e.g. experience level of an agent);
2. *Happenings*: exogenous events (e.g. storm, phone call) can be triggered, as long as they don't have to be explained by the domain model;
3. *Occurence Constraints*: depending on the granularity of the domain modeling, some behaviours can have uncertain effects, that rely on a random draw (the spattering linked to a leak, for instance); DIRECTOR can constrain the occurrence or non-occurrence of such effects.

3.2.3 Plot Selection

To reduce the computational cost, the scenario generation process is split into two phases: first, a plot is selected from a predefined causality model. Then, this plot is instantiated and completed through a planning process.

A plot is a partially ordered graph composed of two types of plot points: events and situations. These plot points are non-instanciated, referring to objects types and not instances. DIRECTOR selects the plots from a causality model (Figure 3) — an AND/OR

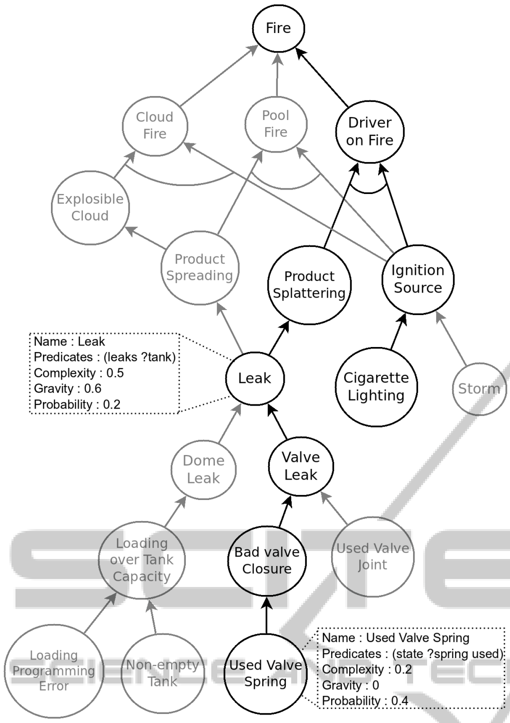

Figure 3: Extract of a causality model and a selected plot.

graph created by risk-analysis experts, which contains causal relationships between possible accidental events and unwanted situations, tagged with their respective complexity, gravity, and relative probabilities. Possible plot graphs are generated by expanding the portion of the causal graph from the goal situation, splitting at every OR gate. These plot graphs are evaluated regarding the situation constraints (desirability of the situations appearing in the graph) and the metric constraints given by TAILOR. One plot is randomly selected between the plots with the best evaluation.

Given the representation of a plot, where:

$p \in P = \langle N_p, A_p \rangle$ is a *plot*, i.e. a set of partially ordered *plot points*

N_p is a set of nodes (*situations* or *events*) n

A_p is a set of causal arcs a

The evaluation functions of currently implemented metric constraints are presented in Table 1.

3.2.4 Plot Completion

Once the plot is selected, the concepts it refers to are instantiated in regard to existing objects and agents in the current state of the world, so that each *plot point* contains a set of predicates that can be used as a goal by a planning system — in our case, as landmarks (Porteous et al., 2010). However, unlike most

Table 1: Metric Constraints.

Type	**Evaluation function**
complexity	$\lvert A_p \rvert * \sum_{n \in N_p} complexity(n)$
gravity	$\max_{n \in N_p} gravity(n)$
believability	$\prod_{n \in N'_p} F(n)$ where N'_p are the leaf nodes of p and F the statistical frequency of n

planning-based systems, the resulting plan is not prescriptive, but serves as a prediction of the simulation behaviour. The aim of the completion phase is thus not to generate the optimal plan between two plot points A and B, but rather to compute the adjustments needed to bring the simulation from situation A to situation B, and predict the autonomous agents actions from situation A given these adjustments.

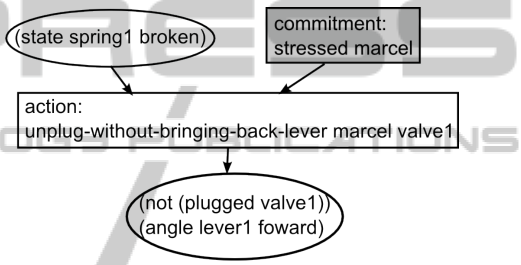

Figure 4: Scenario plan between two plot points.

The use of *actions* as planning operators to compute a plan between A and B would offer no guarantee that the agents would actually follow this plan. On the other hand, using *adjustments* would require to compute a whole scenario prediction each time an operator is tried, to check if this prediction contains situation B. Therefore, two types of operators are used:

1. *Prediction Operators*: they represent the *actions* and *behaviours* that will happen in the simulation. The *actions* operators are generated from the activity model, and framed by preconditions so that, in a given situation, only one operator would be applicable for a given agent (the one that the virtual character would select in the simulation). The simulation *behaviours* are generated from the domain model.
2. *Adjustment Operators*: they correspond to the *commitments*, *happenings* and *occurence-constraints*.

The scenario generation module iterates over the arcs between couples of plot points to replace each one with a plan composed of these five types of operators. Goal situation, current world states and operators are feeded to a external planner (here we use (Hoffmann, 2001)).

3.2.5 Scenario Monitoring and Adjustment

The generated scenario plan is translated into a directed acyclic graph allowing to monitor the execution of actions and behaviours and to trigger the commitments, happenings and occurence-constraints. Indeed, adjustments are not triggered when the scenario is generated, but instead when they are needed, in order for the situation to stay open as long as possible.

4 EXAMPLES

Figure 5 presents two examples of scenario generation for an hazardous matter loading training application (Barot et al., 2011), with two virtual characters, *Marcel* and *Gaston*. In the top example, the learner is a beginner, therefore the set of constraints that DIRECTOR receives as input is:

- $goal = Fire$
- $desirability(DomeLeak) = -0.3$
- $believability \in [0.01, 0.03]$

DIRECTOR first selects a plot, represented by the elliptic nodes in the graph. This plot is instantiated regarding to the environment, and then completed with the planning operators, represented by the rectangular nodes in the graph. The white ones correspond to *prediction* operators, while the grey ones correspond to *adjustment* operators. The unfolding of events in the simulation will be monitored by DIRECTOR, and the planned commitments will be triggered. The second set of constraints is made for a more experienced learner:

- $goal = Fire$
- $desirability(ValveLeak) = -0.8$
- $gravity \in [0.9, 1]$

The generated scenario contains the 3 types of *adjustments*: *commitments*, *happenings* and *occurrence-constraints*.

5 DISCUSSION

We proposed a model of a scenario adaptation process to control character-based complex simulations without limiting user agency or forcing incoherent character or objects behaviours. Our aim was to balance control, emergence and coherence, and indeed:

- the scenario can be controlled by setting different constraints over situations or global metrics;
- scenarios emerge from the domain, activity and causality models without having to be described explicitely, and different scenarios can emerge from different sets of constraints;
- the coherence of behaviours is ensured by the indirect adjustments.

As for now TAILOR can only provide DIRECTOR with situations constraints on situations that appear in the causality model, because the situation constraints filtering is made before the planning phase. Therefore, if an unwanted situation should appear between two selected plot points, the DIRECTOR modules would allow it. Moreover, the linear scenario generation process does not allow to take into account as a selection criteria the number of commitments that are needed for the realization of the scenario. Yet, this should be minimized, as the less commitments are made, the more possibilities are left for adaptation afterwards. An iterative process mixing plot selection and planning phases might solve these problems.

Another prospect concerns the prediction of learner actions. They are currently planned the same way as virtual characters actions, by considering the learner to be a virtual character with an "average" profile. It would be interesting to link the planning process with the learner monitoring module so that the prediction of user activity would be more accurate.

6 CONCLUSIONS

We proposed in this paper a model to dynamically adapt the scenario of a virtual environment in a character-based, emergent approach. This model uses late commitment, exogenous events and occurrence constraints on uncertain consequences to influe on characters and system behaviour without spoiling their coherence. Combinatorial explosion is dealt with by splitting the scenario generation process into two parts: first, the selection of a plot in a predefined plot graph, then the instantiation and completion of this plot according to the current world state.

A first prototype has been implemented inside of the HUMANS software platform and seems to be giving satisfying results in generating coherent scenarios, however only the scenario generation part is functionnal and the execution and monitoring remains yet to be tested. As for now, the relevance and diversity of the generated scenarios is limited by the linearity of the selection, instantiation and planning processes. The next version of the SELDON model will have to deal with a more iterative generation process in order to improve on this point.

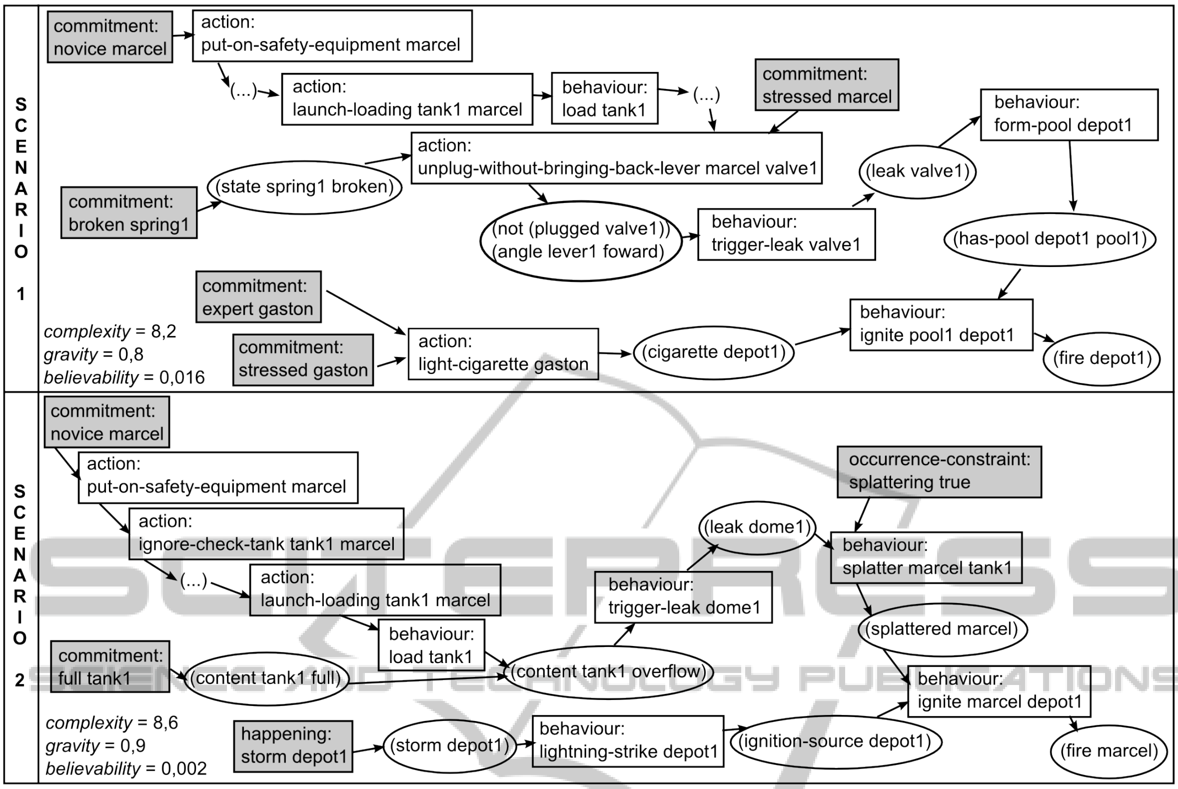

Figure 5: Examples of scenario generation from two sets of constraints.

ACKNOWLEDGEMENTS

The authors would like to thank the PICARDIE region and the European Regional Development Fund (ERDF) for the funding of this work. We also thank Fabrice CAMUS for his work on the field analyses.

REFERENCES

Barot, C., Burkhardt, J.-M., Lourdeaux, D., and Lenne, D. (2011). V3S, a virtual environment for risk management training. In *JVRC11: Joint Virtual Reality Conference of EGVE - EuroVR*, pages 95–102.

Carpentier, K., Lourdeaux, D., and Mouttapa-Thouvenin, I. (2013). Dynamic selection of learning situations in virtual environment. In *ICAART 2013*.

Delmas, G., Champagnat, R., and Augeraud, M. (2007). Plot monitoring for interactive narrative games. In *International conference on Advances in computer entertainment technology - ACE '07*, page 17.

Hoffmann, J. (2001). FF: the Fast-Forward planning system.

Magerko, B., Wray, R. E., Holt, L. S., and Stensrud, B. (2005). Improving interactive training through individualized content and increased engagement. In *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*, volume 2005.

Mollet, N. and Arnaldi, B. (2006). Storytelling in virtual reality for training. In *Technologies for E-Learning and Digital Entertainment*, volume 3942, pages 334–347.

Niehaus, J. and Riedl, M. (2009). Scenario adaptation: An approach to customizing Computer-Based training games and simulations. In *AIED 2009: 14 th International Conference on Artificial Intelligence in Education Workshops Proceedings*, page 89.

Porteous, J., Cavazza, M., and Charles, F. (2010). Applying planning to interactive storytelling: Narrative control using state constraints. *ACM Trans. Intell. Syst. Technol.*, 1(2):10:110:21.

Shawver, D. M. (1997). Virtual actors and avatars in a flexible user-determined-scenario environment. In *IEEE Virtual Reality Annual International Symposium*, page 170171.

Si, M., Marsella, S., and Pynadath, D. (2009). Directorial control in a decision-theoretic framework for interactive narrative. *Interactive Storytelling*, page 221233.

Si, M., Marsella, S., and Pynadath, D. (2010). Importance of well-motivated characters in interactive narratives: an empirical evaluation. In *Third joint conference on Interactive digital storytelling*, page 1625.

Swartjes, I. (2010). *Whose Story Is It Anyway*. PhD thesis, University of Twente.