Emohawk: Searching for a "Good" Emergent Narrative

Cyril Brom, Michal Bída, Jakub Gemrot, Rudolf Kadlec, and Tomáš Plch

Charles University in Prague, Faculty of Mathematics and Physics
Malostranské nám. 2/25, Prague, Czech Republic

Abstract. We report on the progress we have achieved in development of Emohawk, a 3D virtual reality application with an emergent narrative for teaching high-school students and undergraduates the basics of virtual characters control, emotion modelling, and narrative generation. Besides, we present a new methodology, used in Emohawk, for purposeful authoring of emergent narratives of Façade's complexity. The methodology is based on massive automatic search for stories that are appealing to the audience whilst forbidding the unappealing ones *during* the design phase.

1 Introduction

Despite rapid progress in the fields of virtual storytelling (VS) and virtual characters (VC) over the last decade, educational issues remain largely unaddressed in these disciplines [1]. The lack of tools supporting education of beginners is an issue of high priority. In particular, while tools for authoring 3D graphics that can be used for education exist, tools for teaching programming of *behaviour* for VC are almost missing. Many VS authoring toolkits emerged recently [see 2], but they tend to provide very limited or no support for teachers and learners. Many of them are poorly documented and some are not downloadable. To our knowledge, there are only three tools concerned with programming behaviour for VC explicitly supporting education: Netlogo [3], Alice [4], and Pogamut [5]. Netlogo is an entry-level tool for building simple agents and running social simulations, but it does not allow for creation of VCs and possesses no complex 3D environment. Alice's main goal is to introduce basic programming concepts like conditional and loop primitives. The complexity of the built-in 3D world is not comparable to current game engines; the 3D world is "only" a mean to achieve Alice's educational goal. Pogamut is our own toolkit based on Unreal Tournament 2004 (UT) [6]. It has been successfully used in undergraduate as well as high-school education [1], but it is primarily focused on gaming AI audience, with limited use for general VS/VC courses [1].

To address this gap, we have started to develop an educational toolkit Emohawk on top of Pogamut and the UT game. Our premise is that a good introduction into the world of VS/VC is to immerse students in an interesting 3D virtual narrative, which they can co-construct by interaction and *examine its underlying mechanisms at the same time*. Emohawk should assist in teaching: a) basics of 3D VC control, e.g. steering and reactive planning, b) appraisal-driven architectures, c) coordination of multiple characters; i.e. unfolding "atomic bits" of a story, d) composing a story from these "atomic bits". The target audience is students and teachers of general computer

I.A. Iurgel, N. Zagalo, and P. Petta (Eds.): ICIDS 2009, LNCS 5915, pp. 86–91, 2009.

sciences, social sciences, computer games and new media and art. Emohawk is intended for both the high school and university courses, and for both boys and girls. The teaching methodology capitalises on our incremental method presented in [1].

Emohawk features a narrative scenario with three characters controlled by an appraisal-driven architecture. The narrative is partly emergent due to this architecture and also due to natural non-determinism and user interaction. At the same time, it is also partly scripted, for it evolves around some predetermined plot points. Some options are available for authoring such scenarios [e.g. 7, 8]; however, they seem either to be time-consuming during the design phase [8] or to be limiting interaction of the user below the limit acceptable for Emohawk [7]. Thus, we have designed a new authoring method. On a general level, it addresses the issue of integration of emergent narrative (EN) and story-based approach. The main idea is to perform massive automatic search through the story-space *during the design phase* to find subspaces where the EN system may produce appealing stories while forbidding places where the EN system would perform poorly. We employ this strategy due to the unpredictable behavior of EN systems that often produce unappealing stories. The forbidding is realized by the designer by creating constraints layered upon the EN system. In real-time, the provided constrains help to structure the story being unfolded so that the unappealing story arcs are avoided.

The goal of this paper is twofold. First, we present Emohawk; its current state is described in Sec. 2. Second, we overview our authoring method in Sec. 3.

2 Project Emohawk

The Emohawk's content has been being built according to the following principles:

1. the scenario must be interactive and spectator-based at the same time,
2. the scenario must be interesting both for girls and boys,
3. the scenario must feature a story (i.e. a simple narrative arc must be present),
4. the scenario must be short and of Façade's [8] complexity (i.e. tailored to school lessons, with a few characters),
5. the characters must exhibit emotions,
6. there can be a natural language interaction, but all communication has to be handled symbolically, i.e. by body language and emoticons,
7. the setting should be "almost normal", i.e. no "hard core" sci-fi space-ships etc.

The final scenario is as follows: *There is a small city with about thousand inhabitants. At the edge of the city, an emohawk farm lays. An emohawk is a kind of pet having the ability to suck emotions from people and spit (transfer) them on other people. From time to time, an emohawk escapes from the farm.*

There lives Thomas, age 16-19, in the town. Thomas has a girlfriend Barbara and... well... yet another girlfriend, Natalie. The girls naturally don't know about each other. Thomas has just been with Barbara in the cinema. Now, he has to walk her home quickly, since Natalie is waiting for him to be taken to the next movie. Besides, there is an emohawk roaming around.

Now, several things may happen based on whether Thomas and Barbara meet the emohawk or not, and based on their attitudes towards the pet. For example, Barbara

may start to enthuse over the cute emohawk, annoying Thomas and causing Natalie to start searching for Thomas and eventually find him with Barbara. Now, Natalie may start to enthuse over the emohawk as well, or get mad, or something else might happen. Essentially, the author has created the plot as a branching one.

Two modes of interaction with the emohawk's world are being designed. The first is oriented on beginners and the second on advanced students. The first is a game in which a student is a spectator and influences the course of the story by interactions; e.g. switching on/off individual appraisal rules of certain characters or putting some objects into the world, e.g. a cute teddy bear next to Barbara (the object may attract her attention, delaying the normal course of events and changing the story outcome later on). The second mode is programming: students will have a code of a character that they can change easily and run immediately. They will start with making simple concrete changes and gradually proceed to abstract complex assignments.

Technically, Emohawk is being developed as an extension of our Pogamut toolkit [5], which has been already released. Pogamut differs from other projects using UT in featuring a full-fledged integrated development environment and auxiliary libraries facilitating development of characters' control mechanisms.

The prototype of the scenario is already implemented [8]. The characters' action selection is BDI-based. We use ALMA [9], which is based on the OCC theory [10] as the characters emotion model. Our instantiation of ALMA features several complementary pairs of *emotions* and *attitudes* towards other characters/objects. Both the action selection mechanism as well as the emotion model can be influenced by a story manager featuring reactive rules. Emotions are expressed by different animations and the most salient emotion/attitude is represented by coloured bubbles close to a character's head (see [11]).

3 Story Design in Emohawk

User interaction, random fluctuations and the appraisal-driven architecture of characters in Emohawk makes behaviour of the whole application non-deterministic; basically, we have an EN system. At the same time, we have a plot, which should be unfolded. It can and should be unfolded a bit differently each time, but nevertheless constrain what would be otherwise a pure emergent narrative. How to author this hybrid? Louchart et al. [12] suggest that the solution to EN authoring may be for the author to stop thinking "in terms of plots" and to start thinking in terms of "goals, actions and emotions there are, and under which conditions they occur" [12]. This could possibly lead to emergence of interesting episodes of local interactions among VCs and/or the user. Connecting these episodes into an interesting story is then, to a large extent, a responsibility of the user. In our opinion, this approach does not guarantee that a) local interactions are interesting (the result may be a quite accurate computational model of social interaction, which nevertheless features no appealing narrative), b) the user will be able to proceed from one episode to another despite his/her wishes (e.g. the couple will never meet the emohawk for the pet's appraisal-driven architecture will always lead the pet elsewhere, precluding the user to enter a large part of the story).

We reason differently. In every instant, the application *affords* the user a set of actions through which s/he can influence the world. The user anticipates outcomes of these actions and selects from them just one that appeals to him/her most at the given moment. Our opinion is that the author should think in terms of these *possible action outcomes*. The challenge is to afford to the user *appealing* action outcomes for a reasonably large number of contexts the user might encounter. It does not matter whether an outcome pushes the story forward or just colours the current episode in an interesting way (e.g. a user may push Thomas to offer a frog to Natalie); the only important thing is that the outcome is appealing in some way. Here, we actually need both "plot-oriented thinking" as well as "goals & emotions-oriented thinking"; these are unified in our framework. Note that although the whole story can be appealing to the user who co-created it, it may not be appealing to an external observer.

This vague idea materializes in the authoring method having been used in Emohawk as follows. The designer comes up with (a) an ideal of a branching narrative, and (b) with an EN system that produces a space of possible story unfoldments that may or may not be appealing to the audience. Let us call this space *S*. Now, the designer carves a smaller space of acceptable story unfoldments from *S*, so that each acceptable unfoldment approaches the desired ideal, i.e. she/he prunes away a large part of *S* and extends the rest a bit, producing a new space *R*. In other words, the purpose of the EN system is to automatically afford as many actions to the user as possible. The purpose of the reduction *R* is i) to make it impossible to execute actions with uninteresting outcome, and ii) to afford appealing action outcomes in situations in which the EN component does not afford anything interesting.

More precisely, we need the notion of a *dramatic situation* and a *corridor*. The former is a situation that affords naturally multiple interesting action outcomes, either outcomes just colouring the situation or pushing the story forward, e.g. "Barbara and Thomas meet the emohawk". The latter connects two dramatic situations and affords only a few interesting action outcomes. The author should organise the plot around these dramatic situations and corridors, i.e. to create a branching graph. The notion of a branching graph is not new, but we tend to look at it in a new way. We conceive every path of the graph as a representation of an optimal story evolution as foreseen by the author under the assumption of no user interaction. However, due to user actions and natural non-determinism, characters can depart from these optimal paths. Many of these "deviations" can be anticipated by the author, but some of them may be missed. Still, many of these unanticipated "deviating" stories can be interesting. To describe these stories, we need the notion of an *enabled corridor*. An enabled corridor represents situations that do not differ significantly from either a dramatic situation or a corridor (Fig. 1a) and still afford reasonable number of interesting action outcomes.

Now, with the branching graph in hands and with the idea of an EN system (which includes the idea of VCs' architecture, initial goals and parameters), we setup the virtual world, put our VCs into the initial settings, and let the story to evolve deterministically. We tweak the parameters of the VCs to produce a story that accords, at least approximately, with one of the possible paths in the branching graph. Now, we need to define the enabled corridors. Running the scenario multiple times, we investigate what happens when user actions and random fluctuations are taken into account, forbidding stories to evolve into situations with uninteresting action outcomes. Assume we can somehow detect these situations. If such situation is encountered due to a user interaction, we should forbid this interaction *in the design phase* (e.g. there will be no

teddy bear in the simulation). Otherwise, we can implement (in the design phase) a new reactive rule for the story manager that influences (during run-time) the story a moment before the situation is encountered so that the situation is avoided "naturally" (e.g. the emotional state of a character is altered unknown to the user—Fig. 1a). Additionally, an entirely new corridor leading to a new dramatic scene unforeseen by the author can be detected (Fig. 1c).

The problem is that detecting uninteresting situations, in general, is intractable; we can apply only a heuristic approach. Our heuristic was following: we formalised what it means that the story is interesting, based on states and changes of states in VCs' affective variables. This is inspired by Façade [8]. Then, we run the scenario over 300 times (these runs evolved non-deterministically), picked automatically runs that violated the definition of interestingness (around 30% of them) and inspected these runs manually by the designer for uninteresting situations or errors. The whole process cannot be explained here for brevity; it is fully detailed in [11]. Obviously, this approach can be only as good as the formalization is. For instance, there might be scalability problems for larger narratives or narratives that are not affective based.

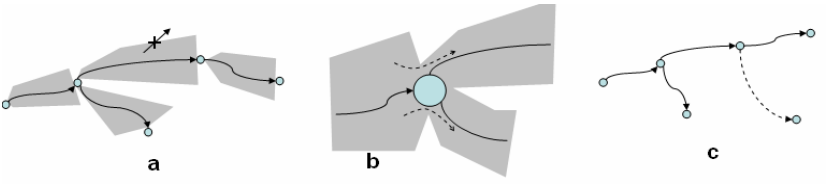

Fig. 1. a) The branching graph with enabled corridors (in grey) and rules forbidding the story to leave an enabled corridor. b) Either the story can follow one of the branches automatically due to the EN component, or the story manager can enforce the story to follow a specific branch. c) During the design phase, a new branch can be discovered (dashed line).

4 Discussion and Conclusion

We presented Emohawk: a 3D virtual storytelling application for teaching basics of the fields of virtual characters and virtual storytelling. The project is an on-going work. The main contribution so far is the new authoring approach we used to create a prototype of the Emohawk's scenario, as detailed in [11]. The approach is based on combining emergent narrative with scripting: the scripted part influences the emergent component by means of reactive rules that are created during the design phase based on the results of multiple runs of the scenario and automated detection of situations that should be avoided during a story evolution. Thus, the problem of combinatorial branching of possible stories is tackled by an off-line search.

Generally, our approach complements other methods of authoring 3D virtual narratives of this complexity, such as [7, 8]. Similarly to these methods, it is not without limitations. First, it seems that it can be scaled only for relatively small narratives with "interestingness" that can be formalised. Second, even for "the right" narratives and even after an extensive search, some uninteresting scenes are likely to be kept and it may happen that a user escapes to a terra incognita by an unpredicted sequence of actions. However, as the problem is intractable, we can not hope for a 100% correct solution: we will *always* have only a heuristic based solution. In our case, the

approach produced both novel dramatic scenes unforeseen by the author and helped to detect uninteresting evolutions of the narrative. This is clearly an improvement comparing both to a pure EN system as well as to a pure scripting.

In general, this paper showed that massive off-line search for appealing stories can be added to the portfolio of authoring approaches used in digital storytelling, though caution is needed.

Acknowledgments. This work was partially supported by the project CZ.2.17/3.1.00/31162 financed by the European Social Fund, the Budget of the Czech Republic and the Municipality of Prague, by Program "Information Society" (under project 1ET100300517, and by the research project MSM0021620838 of the Ministry of Education of the Czech Republic. The name "Emohawk" is inspired by Emohawk: Polymorph II, an episode of Red Dward VI (BBC).

References

1. Brom, C., Gemrot, J., Burkert, O., Kadlec, R., Bída, M.: 3D Immersion in Virtual Agents Education. In: Spierling, U., Szilas, N. (eds.) ICIDS 2008. LNCS, vol. 5334, pp. 59–70. Springer, Heidelberg (2008)
2. Pizzi, D., Cavazza, M.: From Debugging to Authoring: Adapting Productivity Tools to Narrative Content Description. In: Spierling, U., Szilas, N. (eds.) ICIDS 2008. LNCS, vol. 5334, pp. 285–296. Springer, Heidelberg (2008)
3. Wilensky, U.: NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University (1999), http://ccl.northwestern.edu/netlogo/ (26.6.2009)
4. Carnegie Mellon University: Alice (1999-2009), http://www.alice.org/ (10.9.2009)
5. Kadlec, R., Gemrot, J., Burkert, O., Bída, M., Havlíček, J., Brom, C.: Pogamut 2 – A Platform for Fast Development of Virtual Agents' Behaviour. In: CGAMES 2007, pp. 1–5. Univ. La Rochelle (2007), http://artemis.ms.mff.cuni.cz/pogamut (26.6.2009)
6. Epic Megagames: UT 2004, http://www.unrealtournament.com/ (26.6.2009)
7. Aylett, R.S., Louchart, S., Dias, J., Paiva, A., Vala, M.: FearNot! – An Experiment in Emergent Narrative. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 305–316. Springer, Heidelberg (2005)
8. Mateas, M.: Interactive Drama, Art and Artificial Intelligence. PhD thesis. Department of Computer Science. Carnegie Mellon University (2002)
9. Gebhard, P.: ALMA - A Layered Model of Affect. In: Dignum, F., Dignum, V., Koenig, S., Kraus, S., et al. (eds.) AAMAS 2005, pp. 29–36. ACM, New York (2005)
10. Ortony, A., Clore, G.L., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press, Cambridge (1988)
11. Bída, M.: Artificial emotions in computer games. Master thesis. Charles University in Prague, Czech Republic (2009), http://artemis.ms.mff.cuni.cz/emohawk (10.9.2009)
12. Louchart, S., Swartjes, I., Kriegel, M., Aylett, R.: Purposeful authoring for emergent narrative. In: Spierling, U., Szilas, N. (eds.) ICIDS 2008. LNCS, vol. 5334, pp. 273–284. Springer, Heidelberg (2008)