Distributed Drama Management: Beyond Double Appraisal in Emergent Narrative

Allan Weallans, Sandy Louchart, and Ruth Aylett

MACS, Heriot-Watt University, Edinburgh, Scotland EH14 4AS
aw119@hw.ac.uk, {sandy,ruth}@macs.hw.ac.uk

Abstract. In this technical paper, we describe an implementation of Distributed Drama Management (DDM). DDM is a concept which involves synthetic actor agents in an Emergent Narrative scenario acting on both an in-character level, which reflects the concerns of the characters, and an out-of-character level, which reflects the concerns of a storyteller. By selecting the most "dramatically appropriate" action from a set of autonomously proposed actions, Distributed Drama Management aims to retain the benefits of Emergent Narrative such as believability and agility of response to user actions, but attempts to provide a structurally and emotionally consistent experience.

1 Introduction

1.1 Overview

This is a technical paper describing the implementation of a concept known as Distributed Drama Management (DDM). At the core of DDM is the idea that in an interactive digital story facilitated by autonomous agents, those agents must be aware on one level of the story from the perspective of the characters they represent, but also on another level from the perspective of a storyteller.

In traditional, non-interactive stories, characters in and of themselves are unaware of several narrative considerations the author must take into account. For example, characters have no concern for the plot of the story in which they are participants. It is the concern of the author to select her characters' actions such that they serve the plot without appearing unbelievable. However, incorporating such a degree of authorial control into an interactive story is, as described by the Narrative Paradox [1], antithetical to significant breadth of freedom in user interaction. Our approach to resolving this – DDM – is inspired by the role-playing game (RPG) practice of distinguishing between considerations that are *in-character* (IC, which is to say, looking at situations from the perspective of the character the role-player is playing), and considerations that are *out-of-character* (OOC, which is to say looking at situations not from the perspective of the character, but from the perspective of the role-player herself). Therefore, it is appropriate to think of DDM agents more as virtual role-players than as virtual characters.

For the remainder of this section, we will describe DDM at a high level. In the next section, we will examine the related work in the field. In section 3, we will explain the concepts of DDM in more detail, and in section 4 we will describe our current implementation of DDM and provide an example of how it works.

1.2   Distributed Drama Management

There are, broadly speaking, two approaches to Interactive Digital Storytelling. One approach is the plot-centric (also called top-down, plot-centric, or author-centric) approach, while the other is the emergent (also called bottom-up, or character-centric) approach. The relative advantages of these two approaches are well-established, e.g. in [2]. The top-down approach has benefits in structure, pace, and authorial expression, while the bottom up approach is advantageous for narrative coherence and consistency (which contribute to believability) and agility of response to user interaction.

Much study has been made into approaches to bridging this gap. DDM is one such approach. A selection of other approaches will be examined in the next section.

At a high level, the DDM system comprises five components. Characters are represented by virtual actors which actually account for two of the components: the **Character Layer**, which is responsible for simulating the character according to its own beliefs, desires, intentions and emotional state, and the **Actor Layer**, which is responsible for mediating the possible actions generated by the Character Layer in terms of their dramatic appropriateness. Dramatic appropriateness, in DDM, is contingent on how the action will affect the character the user is role-playing, which is represented in DDM by a special agent called the **Virtual User**, which does not act in itself, but represents the beliefs, desires and emotional state of the user's character. The Actor Layer sends proposed actions, along with their simulated emotional impact, to a **Drama Manager**, which, unlike traditional managers, does not exert directorial control over the agents, but merely compares all proposed actions from all actors against the target emotional trajectory, and then selects and authorises the one with the best fit. The target emotional trajectory is specified by the final component, the **Story Specification**, which is an authorially-produced document describing the story at a high level of abstraction as a sequence of 'episodes', each with its own emotional target.

The process of DDM begins with the Character Layers generating a set of all actions it can perform that are consistent with both its goals and current emotional state. Actions are defined as either steps taken by the character in pursuit of a goal (deliberative actions) or reactions to an event (reactive actions). The Actor layer then *simulates* the effect of each of these actions on the Virtual User – no actions are executed in the world at this time – and proposes those actions that are closest to the emotional target to the other agents, which then simulate the action they would subsequently perform. This creates a sequence of two actions, representing not only the proposed action, but also what can immediately follow. The Actor Layer then simulates the emotional impact of this sequence on the Virtual User and notifies the Drama Manager of the sequence and its predicted emotional impacts. Once the Drama
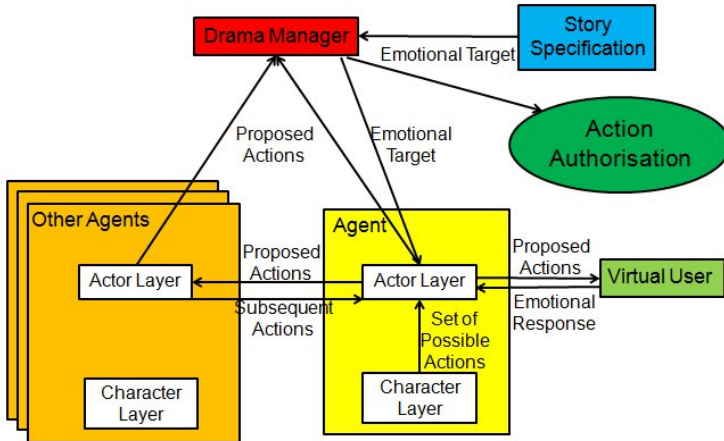
Fig. 1. A high level diagram showing the different components of DDM and the relationships between them

Manager has received all sequences from all agents, it compares all of the emotional impact sequences to the emotional target provided by the Story Specification, and selects the one with the closest match. The initiating action of that sequence is then authorised for execution in the story world, and the agent, upon receiving that authorisation, executes that action.

2   Related Work

DDM makes use of FAtiMA (Fearnot AffecTIve Mind Architecture) agents, and so *FearNot!* [3] is significant related work. *FearNot!* is a character-based system relying on the properties of Emergent Narrative to allow a user to participate in a story in which a child is being bullied. *FearNot!* is notable for its affectively-driven agent architecture incorporating continuous planning and Ortony, Clore and Collins' (OCC) emotion taxonomy [4], resulting in sophisticated autonomous agents. It is our belief, as discussed in section 3.1, that FAtiMA agents address one component of character, but there are other components of character that are not addressed, and it is into this gap that DDM fits.

The Virtual Storyteller [5] represents Swartjes' work on using improvisational theatre as a model for Emergent Narrative. Swartjes notes that, like improvisational actors, Emergent Narrative agents should be aware of and act on both in-character and out-of-character concerns. This is our reasoning behind the use of a Character Layer and an Actor Layer in DDM.

On the theme of improvisational theatre, Magerko and Baumer's *Digital Improv Project* [6] aims to create improvisational agents on the basis of "offers and responses". Agents can make 'offers' to attempt to advance the story, which other agents can subsequently either accept and augment, accept in a qualified way, or

block, while presenting an alternative. DDM is similar to the "accept and augment" response, but agents do not accept the proposal themselves; rather, their augmentation is more hypothetical – the Actor Layer says what the agent would do if the proposed event occurs (*i.e.* is selected by the Drama Manager).

Riedl and Young's *Fabulist* [7] is a narrative generation engine that creates a plan (which is to say, a sequence of character actions) to achieve goals specified by the author, and then reasons about whether or not those actions appear believable based on the intentions of the acting characters, back-tracking and replanning if any of the actions appear unbelievable. In principle, DDM is the inverse of this – it begins with believable actions, and then selects the action that best matches the story needs.

Szilas' *IDTension* [8] represents an attempt to simulate "the laws of narrative". It uses narrative logic to calculate all the possible actions in the story world at a given time, and then a narrative sequencer to estimate the impact each potential action would have on a 'user model', to determine which action would be most valuable.

Persu, described by Figueiredo and Paiva in [9] is an architecture for systematically deploying persuasive content in order to encourage a user of an IDS system to adopt goals and perform actions that serve authorial goals for the story. While DDM is not concerned with the user performing actions that serve authorial goals, we believe there is scope for adapting *Persu*'s methods to persuade the user to adopt goals that are appealing to the playable character, thus reinforcing the relationship between user and character. This, however, remains potential future work for DDM.

Finally, it is necessary to compare the Drama Manager of DDM with search-based drama management such as declarative optimization-based drama management discussed by Nelson and Mateas in [10]. Search-based drama management is directive and interventionist: it searches for actions that fulfil authorial goals. DDM's Drama Manager, by contrast, is opportunistic: it authorises and denies emergent actions depending on their suitability to the authorial goals.

3   Approach

3.1   The Character/Actor Distinction

Firstly, we believe that the weaknesses of Emergent Narrative can be largely attributed to the fact that EN agents are focused on providing a high degree of simulative fidelity, which is to say that they are "virtual people". With reference to Scholes, Kellogg and Phelan [11], characters in a narrative are (to varying relative degrees depending on cultural norms and on individual stories): *aesthetic* in that they serve the plot of the narrative, *illustrative* in that they represent or symbolise certain ideas or themes, and *mimetic* in that they simulate human beings. Characters are self-aware only insofar as they are aware of their mimetic components; their aesthetic and illustrative components exist outside of the fictional world, such that only the author and reader may be aware of them.

Due to the focus on simulative fidelity, the current state of EN agents emphasises the mimetic component of character, and as such, these agents are unaware of their

narrative responsibilities. EN agents will act believably, but it is not guaranteed that these agents will perform dramatically interesting actions, much less that they will perform a sequence of dramatically interesting actions, and less still that all agents will do so in a form structurally recognisable as a story. One attempt to address this weakness was Double Appraisal [12] and DDM is an extension of this work. In the Double Appraisal approach, characters did not themselves select actions to perform, but generated a set of possible actions to perform. Because this set of actions was generated by the character agent, all actions in the set were guaranteed to be believable, retaining the benefit of emergence. The emotional impact of each action in the set on other agents was simulated before committing the action to the story. On the basis that emotional impact is a surrogate for drama, the action with the largest emotional impact on the other characters (and, therefore, the "most dramatic" action) was selected.

The weaknesses of Double Appraisal were twofold. Firstly, Double Appraisal selected the action with the largest emotional impact, but did not consider which emotion was being impacted. Under Double Appraisal, an action that made a character very distressed and an action that made a character very happy might be considered of equal dramatic value. Secondly, with reference to Aristotelian structure [13], such as rising action, climax and falling action, the "most dramatic" actions should occur only at the climax, and not at every point in the story. It should be noted that the weaknesses we identify with Double Appraisal here are related to how it was applied, and not with the premise of using Theory of Mind to project the emotional impact of an action, and use that in action-selection. Indeed, we are using this premise in DDM.

It is for these reasons that DDM agents use a two-layered approach, comprising a Character Layer, which is, essentially, the current state of EN agents, and is responsible for the mimetic component, and an Actor Layer, which is responsible for handling narrative concerns of which the characters are themselves unaware, which is to say the aesthetic and illustrative components. In the implementation described in this paper, we are concerned only with the aesthetic component, but the illustrative component remains a promising avenue for future research. An analogy can be made here with the role-playing game (RPG) practice of distinguishing between in-character (IC) thoughts, speech and actions, and out-of-character (OOC) thoughts, speech and actions, the significance of which to interactive digital storytelling was highlighted by Swartjes in [5].

3.2   The Virtual User

We believe that an interactive Emergent Narrative approach should necessarily consider the role of the user. Pursuant to this, DDM incorporates a Virtual User, which is a special class of agent that does not, in itself, act, but is responsible for modelling the emotional state of the playable character (PC). In this, we are influenced by both the effect of a protagonist's emotional state on a reader in a non-interactive story (*e.g.* work by Komeda and Kusumi [14]) and by the relationship between user and playable character in an interactive work such as a digital game

(*e.g.* the work of Keogh [15]). It is necessary to consider the relationship between user and protagonist from both an interactive and non-interactive perspective because it is well-established [16] that a user of an interactive digital story embodies two roles: that of a spectator (or audience), and that of a participant (or actor).

The Virtual User is the 'target' for action selection, which is to say that while Double Appraisal selected actions based on the emotional impact they had on any character, in DDM, actions are selected based on the emotional impact they will have specifically on the Virtual User. Because a change in emotional state of the playable character affects the user's experience, we believe that manipulating how the Virtual User is affected affords some control over the user experience.

Furthermore, actions are simulated one step ahead, which is to say that as well as simulating the emotional impact of the proposed action, we also simulate the emotional impact of a hypothetical subsequent action by any other agent. This allows us to evaluate the emotional appropriateness of the action not only in terms of the action in itself, but also in terms of any immediate consequences.

In this implementation of DDM, the Virtual User represents a playable character as written by the author. Toby Gard [17] makes a distinction between 'avatar' and 'actor' type playable characters, where 'avatars' possess no individual personality beyond that projected onto them by the user, and 'actors' possess a fully-realised individual personality created by the author. The DDM Virtual User currently represents what Gard would describe as an 'actor', and necessarily so, because a character needs a personality before the emotional impact of an action on that character can be simulated. However, while the nature of DDM creates difficulties for experiences that allow user-defined characters, we do anticipate future work involving making the Virtual User adaptive to the user, using a pre-authored character only as a baseline.

3.3   Authorial Considerations

Experience has shown that authorship of Emergent Narratives is difficult. With reference to the Narrative Paradox described in [1], authoring Emergent Narrative necessarily involves relinquishing some authorial control as compared to plot-centric approaches [3]. We, however, believe that previous approaches to Emergent Narrative have required the author to relinquish too much control; that there are certain elements the author can control without interfering with the emergent properties of the scenario.

To this end, Distributed Drama Management incorporates a Story Specification, which describes the story at a high level of abstraction, as defined by the author. This comprises a sequence of episodes, which are analogous to the 'functions' of Vladimir Propp [18]. These episodes are defined at a high enough level of abstraction that they do not interfere with emergence, and provide a starting point for a narrative structure. In this implementation, each episode specifies an *emotional target* for the Virtual User, which is to say a given emotion or set of emotions that the Virtual User should experience, as well as a target level for those emotions. It is against this emotional target that "emotional appropriateness" of an action described above with reference to the Virtual User is evaluated.

3.4   The Drama Manager

Finally, DDM does make use of a Drama Manager. Unlike traditional drama managers, DDM's Drama Manager does not direct the agents to perform any action directly. Instead, it receives all the proposed actions from the agents and makes an executive decision as to which one most closely suits the needs of the story. No agent can act without the authorisation of the Drama Manager, but any actions the Drama Manager authorises will be an action the acting agent proposed, and therefore will retain the properties of emergence.

4   Implementation

4.1   Architecture

The DDM system is built on top of the FAtiMA agent architecture [3]. The Character Layer is, itself, a lightly modified version of a FAtiMA agent, the major difference being that it does not select actions, but generates sets of actions the character can perform. These actions are generated by the reactive and deliberative components of the agent and sent to the Actor Layer. The Actor Layer is responsible for simulating the emotional impact of an action on the Virtual User, for communicating with the Actor Layers of other agents to learn subsequent actions, and for proposing sequences
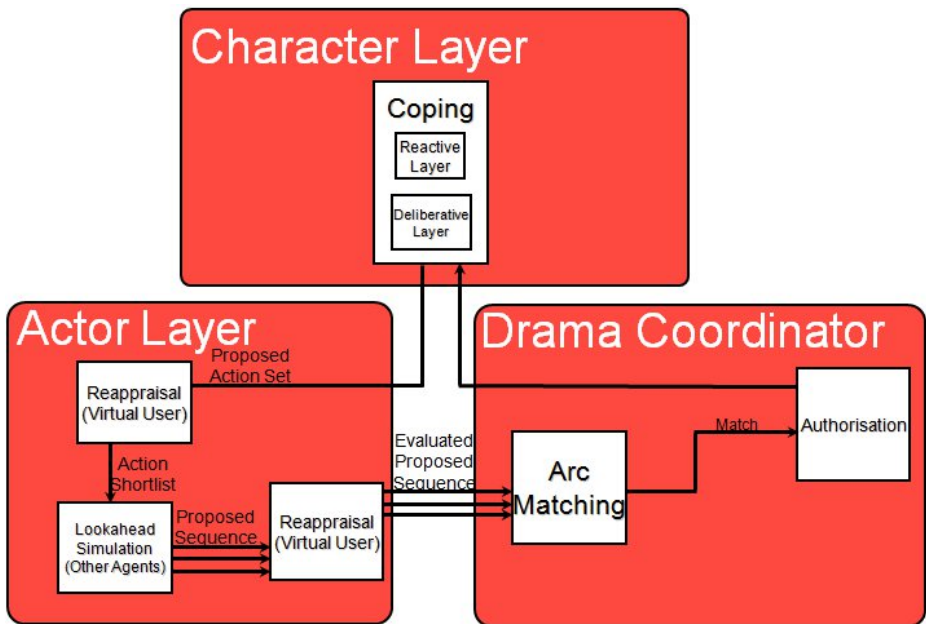

Fig. 2. Diagram of the DDM architecture

of actions to the Drama Manager. The Drama Manager is then responsible for determining which sequence is the best fit for the author-specified emotional target, and authorising the action that initiates that sequence. This authorisation is sent back to the appropriate agent, which then executes it. All communication occurs via the world simulator in the form of command-messages, which are invisible to the user, but can be perceived by agents and handled by the Actor Layer.

4.2   Example

To illustrate the DDM process, we have prepared a small example. It is to be noted that this example exists purely for illustrative purposes: It is non-interactive, and while it is necessary to use a simple example to illustrate how DDM works, this makes it too simple to illustrate why DDM is useful.

Our scenario comprises three characters: Batman, Superman and The Game Master. The Game Master is superhumanly good at videogames. He is bored of playing games, because none of them can offer him a worthwhile challenge, and wants to use his special abilities as a superhero. The scenario involves Batman and Superman conducting a job interview to determine if The Game Master is a suitable candidate to be a superhero.

In this scenario, Batman and Superman are represented by DDM agents, and The Game Master is represented by a regular FAtiMA agent, in the stead of the user. The Game Master's agent takes on Virtual User responsibilities, such that the DDM system is trying to affect the emotional state of The Game Master's agent.In the following subsection, wherein we describe the decision-making process of the DDM system, we will illustrate each step by describing what happens in this scenario.

4.3   Decision-Making Process

The first step in the Distributed Drama Management action-selection process is, necessarily, to create a set of actions from which to select. Because merely selecting from the agent's action repertoire would negate the benefits of using autonomous agents, firstly the agent must, on the Character Layer, decide which actions are appropriate in the current context from both a situational perspective and in terms of the agent's internal emotional state.

Actions can come from two components of a FAtiMA agent: The reactive layer is responsible for reactions that are a direct response to an event, while the deliberative layer is responsible for ultimately selecting actions that an agent undertakes in pursuit of its goals. To generate the set of potential actions for DDM action selection, we must first collate the potential actions from these two sources.

We have considered that reactive and deliberative actions should be handled differently by Distributed Drama Management. Absent concrete conclusions, however, this difference is not covered in this implementation and is a subject for future work.

Generating a set of actions from the reactive layer involves examining the agent's *action tendencies* – which is to say, what the agent will do under certain circumstances – and to identify which of those circumstances are currently true, and

thus, which action tendencies are active. However, it does not suffice to merely generate a set of actions from the deliberative layer, as this would only include actions associated with the currently active *intention*, regardless of if the character may have other intentions. As such, on the deliberative layer we do not generate a set of actions, but a set of intentions: Our set of potential actions from the deliberative layer is the union of the sets of potential actions for all potential intentions. Having generated our two sets of potential actions, reactive and deliberative, these are passed up to the Actor Layer of the agent.

Example: Superman has no reactive actions to perform, as there is not yet anything to react to. For active pursuit goals, Superman's goal is to make a decision on whether or not to hire The Game Master, which requires asking questions in four categories. Superman can choose to ask a hard question or an easy question in each category. This provides eight possible actions Superman can perform at this time.

The Actor Layer receives from the Drama Manager the target emotional change for the action. This target comes from the episode specification, and comprises the emotions being targeted, and whether the actions affecting those emotions should be low, medium or high impact.

Example: We have decided that we want The Game Master to experience a mildly stressful interview, so our scenario specification reflects that we want actions to evoke an emotion of Distress with an intensity of 3.0 (emotional intensities in FAtiMA existing on a scale between 0.0 and 10.0). This is the emotion that the Drama Manager and the Actor Layers of all agents are trying to match.

The Actor Layer then performs Double Appraisal to discover the emotional impact that each action in its set of potential actions would have on the Virtual User. It selects a number of these (defined by a constant in the code) to propose. The selection is done by calculating the 'distance' between the action's emotional impact on the virtual user and the emotional target, and selecting those with the lowest 'distance'.

Example: The Game Master has varying attitudes about the questions he can be asked. This is reflected in the desirability of these actions for The Game Master, which creates distress (or joy) in FAtiMA. By simulating the potential actions on the Virtual User, Superman's Actor Layer discovers that the three actions that will come closest to evoking a distress of 3.0 are a hard question about attitude, an easy question about teamwork, and an easy question about background, as seen in Table 1. These become Superman's actions for proposal.

Table 1. Superman's potential actions

Action	Emotion
question_powers_easy	Joy 6.0
question_powers_hard	Joy 3.0
question_attitude_easy	Joy 3.0
question_attitude_hard	**Distress 3.0**
question_teamwork_easy	**Distress 3.0**
question_teamwork_hard	Distress 6.0
question_background_easy	**Distress 3.0**
question_background_hard	Distress 6.0

Having evaluated all actions in the set of potential actions in this way, the agent sends a signal into the world about each action that it is proposing. This signal specifies the ID of the proposal, subject (*i.e.* the agent itself), the action, and the target (*i.e.* on whom the action will be performed).

Other agents then simulate what they would do if the proposing agent performed that action. While this involves action selection, using Distributed Drama Management action selection at this point would lead to infinite recursion, so these agents use FAtiMA's default action-selection mechanism.

Having decided what they would do after the proposed action, agents use Double Appraisal to simulate what effect their second action would have on the Virtual User if the proposed action had already been performed. Having simulated this emotional response, the agents send a signal out into the world specifying the ID of the proposed action and the emotion type and intensity of the active emotion elicited in the Virtual User by their follow-up action.

Example: Batman's Actor Layer receives Superman's three actions for proposal, and simulates what Batman would do next (without considering impact on the user). At this point, in this simple scenario, there are only two actions Batman can perform that would make sense: to soften the question or to harden the question (which evoke mild joy and mild distress respectively in The Game Master). The simulation, by FAtiMA's existing action-selection method, shows that Batman would soften a hard question about attitude (evoking joy at 3.0), harden an easy question about teamwork (evoking distress at 3.0), and soften an easy question about background (evoking joy at 3.0).

When an agent perceives a response to one of its own proposals, the agent creates a sequence of emotional impacts on the Virtual User, beginning with the emotional impact of the appropriate proposed action and subsequently the emotional impact reported by the other agent. The agent sends this sequence out into the world. If the Actor Layer has received follow-up emotional impacts from all other agents, it removes the proposed action from the list of actions for proposal. Once that list is empty, the agent sends a message into the world signifying that its proposals have been exhausted.

Upon receiving a proposal, the Drama Manager's first action is to classify the emotions being proposed. This may be necessary if the author has specified that the emotional target is on one of a number of emotions rather than on a specific emotion, *e.g.* if the author has specified emotions of negative valence as the target. Our example, however, is targeting one specific emotion.

Once the Drama Manager perceives signals from all agents that their proposals have been exhausted, it can begin to compare them. This involves ranking the proposed sequences according to how well they fit the emotional target given by the episode specification.

We have developed two ways of evaluating this fit: attempting to reach a target level of emotion over the course of the episode, or attempting to maintain a target rate of change. The former involves estimating how many actions will occur in the episode and dividing the target level by that number: this gives the target change for any given action. The latter involves comparing the angle between the action's

emotional impact and the horizontal with the angle between the emotional target and the horizontal (where the horizontal axis represents time and the vertical axis represents intensity).

Example: Table 2 shows the emotional trajectories of the sequences initiated by Superman's shortlisted actions. The easy question about teamwork is ranked highest, because it most closely matches the target trajectory. It also ranks higher than any of the arcs Batman's Actor Layer proposed in the meantime, and so is the action that becomes selected.

Table 2. The proposed action sequences

Superman Action	Emotion	Batman Action	Emotion
question_background_hard	Distress 3.0	soften_question	Joy 3.0
question_teamwork_easy	**Distress 3.0**	**harden_question**	**Distress 3.0**
question_background_easy	Distress 3.0	soften_question	Joy 3.0

Having ranked all the proposed sequences according to their fit to the emotional target, the Drama Manager selects the best one, and sends a signal into the world authorising the initiating action of the sequence by ID.

Upon perceiving an authorisation notification, each agent checks whether the authorised ID belongs to it. If it does, the Actor Layer retrieves the action associated with that ID and sends it for execution in the world. Then all agents clear their proposed action maps in preparation for the next round.

5   Conclusion

The current implementation of DDM concentrates on bringing pace, structure and a coherent emotional tone to Emergent Narrative by selecting the most "dramatically appropriate" action from the set of potential believable actions at any given time.

There is still potential for future development of the DDM concept. Such work may include: adaptivity of the Virtual User to a user's specific interpretations of the character and preferences; authoring tools to assist the author in scenario development, and allowing the drama manager to trigger non-character events (i.e. what Chatman [19] calls 'happenings') if no character actions are suitable.

In the near future, however, we intend to evaluate DDM on two evaluative tracks: user evaluation and author evaluation. For user evaluation, we are interested in whether participating in a narrative experience using DDM has any effect on engagement with the story, using models of engagement such as that described by Busselle and Bilandzic [20]. For author evaluation, we are interested in whether, by providing some very high-level control over the course of the story, it is easier to define a scenario for DDM than for Double Appraisal.

References

1. Aylett, R.S.: Emergent Narrative, Social Immersion and "Storification". In: Procs. Narrative Interaction for Learning Environments (NILE 2000), Edinburgh, UK (2000)
2. Sobral, D., Machado, I., Paiva, A.C.R.: Managing Authorship in Plot Conduction. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2003. LNCS, vol. 2897, pp. 57–64. Springer, Heidelberg (2003)
3. Aylett, R.S., Louchart, S., Dias, J., Paiva, A.C.R., 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)
4. Ortony, A., Clore, G., Collins, A.: The Cognitive Structure of Emotions. Cambridge University Press (1988)
5. Swartjes, I., Theune, M.: The Virtual Storyteller: Story Generation by Simulation. In: Procs. 20th Belgian-Netherlands Conference on Artificial Intelligence (BNAIC 2008), Enschede, Netherlands (2008)
6. Magerko, B., Fiesler, C., Baumer, A.: Fuzzy Micro-Agents for Interactive Narrative. In: Procs. 6th Annual AI and Interactive Digital Entertainment Conference (AIIDE 2010), Palo Alto, CA (2010)
7. Riedl, M.O., Young, R.M.: Character-Focused Narrative Generation for Execution in Virtual Worlds. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2003. LNCS, vol. 2897, pp. 47–56. Springer, Heidelberg (2003)
8. Szilas, N.: IDTension: A Narrative Engine for Interactive Drama. In: Procs.: Technologies for Interactive Digital Storytelling and Entertainment 2003 (TIDSE 2003), Darmstadt, Germany (2003)
9. Figueiredo, R., Paiva, A.: Persu – An Architecture to Apply Persuasion in Interactive Storytelling. In: Procs. 8th Int'l Conf. on Advances in Computer Entertainment (ACE 2011). ACM, Lisbon (2011)
10. Nelson, M.J., Mateas, M.: Another Look at Search-based Drama Management (Short paper). In: Procs. 7th Int'l Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 2008), Estoril, Portugal (2008)
11. Scholes, R., Phelan, J., Kellogg, R.: The Nature of Narrative, Fortieth Anniversary Edition. Cambridge University Press (2006)
12. Louchart, S.: Emergent Narrative – Towards a Narrative Theory of Virtual Reality, PhD Thesis (2007)
13. Aristotle, Poetics (Tr. by Butcher, S.H.. (c.350 B.C.E.), http://classics.mit.edu/Aristotle/poetics.html (retrieved from on June 17, 2012)
14. Komeda, H., Kusumi, T.: The Effect of a Protagonist's Emotional Shift on Situation Model Construction. Memory and Cognition 34(7), 1548–1556 (2006)
15. Keogh, B.: Partners in Crime: The Relationship Between the Playable Character and the Videogame Player, Hons. Thesis (2011)
16. Frome, J.: Eight Ways Videogames Generate Emotion. In: Procs. Situated Play, Digital Games Research Association 2007 Conference (DiGRA 2007), Tokyo, Japan (2007)
17. Gard, T.: Building Character. Gamasutra (2000)
18. Propp, V.: Morphology of the Folktale, 2nd edn. University of Texas Press (1968)
19. Chatman, S.: Story and Discourse: Narrative Structure in Fiction and Film. Cornell University Press (1978)
20. Busselle, R., Bilandzic, H.: Measuring Narrative Engagement. In: Media Psychology, vol. 12(4). Routledge (2009)