Implementing drama management for improved player agency in interactive storytelling

Christopher McEvoy

David J. King

This is the published version of the paper presented at the GAME-ON'2019, 20th International Conference on Intelligent Games and Simulation, 18-20 September 2019, Breda, Netherlands

It is deposited here with kind permission of EUROSIS

McEvoy, S. & King, D.J. (2019) 'Implementing drama management for improved player agency in interactive storytelling', in R Grigg (ed.), 20th International Conference on Intelligent Games and Simulation: GAME-ON'2019, EUROSIS, pp. 63-66.

The full proceedings will be available from :

https://www.eurosis.org/cms/index.php?q=taxonomy/term/58#GAMEON

Implementing Drama Management for Improved Player Agency in Interactive Storytelling

Christopher McEvoy and David King
School of Design and Informatics
University of Abertay, Dundee
Bell St, Dundee
United Kingdom
E-mail: cmcevoy623@outlook.com

KEYWORDS
Drama Management, Intelligent Agent, Player Modelling, Heuristic Search Planning, Graph Theory.

ABSTRACT

With increases in complexity of graphics in video games, there exists a need to increase the complexity of game world narratives so that players feel they are an active part of an unfolding story, influenced by their actions and behaviours. Drama Management (DM) systems offer an attempt to facilitate this but are an area in need of further exploration for application in real-time narrative games. The aim of the project is to develop a prototype DM system for a real-time game that improves player agency and to analyse the effectiveness of the chosen techniques. An application was developed consisting of a 3D interactive environment, a possibility space of narrative plot points, and an Intelligent Agent that branches the story based on a Player Model, using Heuristic Search Planning. It was determined that the possibility space design has a major role in the application's effectiveness to invoke agency within players. The sense of agency can also be improved by combining the developed framework with additional extensions. This project determined that Drama Management systems are a viable method of improving the complexity of a narrative's discourse to promote player agency, but also require careful design alongside suitable algorithmic techniques to be fully effective.

INTRODUCTION

Game AI has been continually improving to match the complexity of computer graphics, to present players with believable and adaptive agents. In the context of narrative games, a similar advancement in the complexity of an interactive discourse, can improve the perception of player agency.

Agency is the degree of influence a player-character's actions have on the state of the game world. This can add to the enjoyment of games as players feel their choices matter.

Drama Management Systems are a computational approach to increasing narrative complexity to improve agency. DM systems use an Intelligent Agent (IA) that decides how to navigate a narrative possibility space defined by the designer.

Storytelling systems fall within two main classes: Emergent Narrative systems, and Drama Management systems. Where Emergent Narrative systems are simulations constructed from Intelligent Agents, Drama Management systems use a single Intelligent Agent, the Drama Manager, to monitor the game world and drive the authored story forward based on the player's actions (Reidl et al. 2012).

Individual approaches may borrow or combine traits from either class. A specific implementation may be considered to fall on a spectrum between prioritising player spontaneity and enforcing an authored narrative arc (Martens et al. 2017). This manifests as a trade-off between designer intent and player agency, which can become a problem. Compelling narratives can effectively balance these attributes.

Emergent Narrative systems are rich simulations underlying the game world that seek to generate unique stories through Intelligent Agents. Emergent Narrative systems have previously incorporated elements of DM systems in their design, creating a hybrid between the two system types. However, conventional DM systems are typically concerned with the structured ordering of authored story content rather than that of generated content. In such systems, a Drama Manager is used to perform the story planning. A Drama Manager is typically a single IA that makes decisions based on logical criteria and one or more additional metrics.

CURRENT TECHNIQUES

While DM systems vary in methodology, they all operate on similar principles to achieve different, but not dissimilar, goals. Events in a linear narrative can be called plot points, which are arranged sequentially. In non-linear games there can exist a state-space of plot points. DM systems use various techniques for constructing or ordering narrative arcs between plot points.

Many systems define plot points with two attributes: preconditions and effects. Preconditions describe what is required for the plot point to be accessible, and effects describe what changes are made to the story world by the plot point. These preconditions and effects connect plot points into an acyclic directed graph of nodes, and this format is used when planning the plot discourse.

DM systems navigate this directed graph but need at least one other module to inform the decision-making, where a module is a separate component that contributes to the overall system. Using information from its other modules, the DM can plan so that future problems can be avoided or to further enhance the story experience by selecting the most preferable path.

To select a plot point from a series of possibilities, many DM systems use a form of Heuristic Search Planning (HSP) to evaluate the best plot point, or trajectory, for a given situation. The heuristic can be calculated based on several attributes usually returned by another module to the DM.

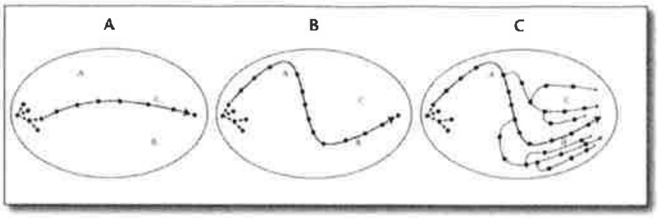

Figure 1: Visualisation of Trajectory Space as a Directed Graph

Figure 1 above (Roberts et al. 2011) is a visualization of trajectory space, as a directed graph, where each dot represents a plot point. There can exist a very large number of possible trajectories that, based on player interaction, are guided towards desirable plot points. Example A shows one possible path through the possibility space. Example B shows how a system can direct the path towards more favourable plot points and away from unfavourable points. Finally, example C shows how an interactive system with user input can lead to branches that navigate the space differently. HSP can be used to decide how favourable points are, and the plot can be branched using the heuristic and actions taken by the player.

The heuristic used to select plot points can be based on output from another module that communicates with the DM. Previous approaches have calculated the heuristic using a plot progression model, while other approaches use various forms of player modelling (PM). Player modelling has been used to model several different metrics through which a DM system can decide the possible future trajectories. DM systems incorporating player modelling attempt to model the player's state explicitly and shape the narrative specifically to influence it (Hernandez et al. 2014).

Player models have been based on a player's predicted emotional state. Others have been based on their predicted playstyle. Some approaches use the playstyle to infer a predicted player goal, using said predicted goal to predict the player's next set of actions, which is then used to change the state of the world in preparation for those actions.

For a playstyle prediction model, one method is to annotate player actions with a set of weights to several playstyle classes created by the designer (Weyhrauch et al. 1997). This annotation is to inform the system of what playstyle the player is likely to be using, to then infer predicted goals. The ordering of actions can then change the weights of the annotations, making a series of actions contribute more to one or more playstyles.

In essence, the way a player interacts with the world is used to determine what they are trying to accomplish, allowing the DM to consult the plot progression model and/or other metrics and use the result to decide what to do. The difficulty of this method is the balancing of the weights and player model algorithm to improve the accuracy of prediction. Additionally, the earlier predictions of player goals are more likely to be incorrect but improve in accuracy with a greater number of actions taken. This approach shares similar principles as reinforcement learning (RL), without the training results being retained.

Previous research and applications in Drama Management have involved the use of Adversarial Search and Partial-Order Causal Link (POCL) planning to structure a dramatic arc while avoiding problems (Roberts et al. 2011), Case Based Reasoning (CBR) to learn how interested participants are in different sections of the plot (Sharma et al. 2010), and player modelling for suspense (Reidl et al. 2011) to name a few. Additionally, natural language processing has been used in several applications with text-based interfaces. Most notably, *Façade* (Procedural Arts, 2005) allows players to communicate with the game's Intelligent Agent Non-Playable Characters (NPCs) using natural language through text.

Other approaches have used DM systems in conjunction with Intelligent Agent NPCs to add depth to an authored narrative, or to facilitate improvisation within a partially generated narrative space. One research example of such an approach is a Distributed Drama Management (DDM) system (Weallans, 2012) which seeks to retain the Emergent Narrative system benefits of believability and improvisation while still providing emotional and structural consistency.

METHODOLOGY

The created solution consists of a real-time 3D environment with limited methods of player interaction, a narrative possibility space represented by a directed acyclic graph and the DM system itself. The possibility space consists of plot points that have preconditions and effects.

Logical Structure

The Boolean condition of the precondition logic can be more complex than a simple 'AND' or 'OR' statement. There may be many conditions required and many logical operators. This project solves this problem by creating a method of defining condition logic, such that statements of any length and combination of ANDs and ORs can be specified. This project refers to the created solution as a 'dynamic Boolean matrix'.

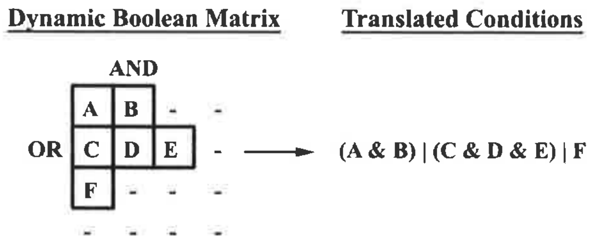

Figure 2: Visualisation of the Representation of Plot Point Precondition Logic

Figure 2 above shows that plot points in a row are ANDed together, and the result of each row is ORed with all other rows. The figure shows how this translates to a non-trivial Boolean statement. The matrix does not incorporate NOT operators as these are represented by the effects.

Player Modelling

The DM uses this logical structure to navigate the possibility space, and decides, out of those with met conditions, which plot points should be accessible in the world. This is determined by Heuristic Search Planning. The heuristic is a similarity measure between plot points and a Player Model, both of which have a set of weights. Plot points have a set of designer-annotated weights, used for HSP, and connected narrative text that is displayed when the point is fired. Once fired, plot point weights are used to update the Player Model and the weights of future points in the same branch as the fired point are also updated to improve their similarity to the new PM.

$$W_n = W_o + ((W_a - W_o) \bullet C_w + (W_a - m) \bullet C_m) \quad (1)$$

The above equation (1) adds a calculated value to the old PM weight to output an updated weight; where W_n is the new weight, W_o is the old weight, W_a is the adjusting weight, m is the constant mid value of 0.5, C_w is a weight constant and C_m is the mid line constant.

The increase value is generated using the difference between the old PM weight and the weight of the fired plot point, defined here as the 'adjusting weight', the difference between the adjusting weight and the 'mid value'. There is also a 'weight constant' and a 'mid line constant' which are used to determine the influence that these differences have on the incremental value.

The DM uses the possibility space structure and Player Model to decide which points to remove from the world, or prevent, and which points to place in the world. The weights of the PM, and of future plot points in the same branch, are updated after each point is fired.

Information Replacement

Information Replacement is the main feature of the Drama Management System. During this process, the IA evaluates what plot points are logically viable, with satisfied precondition logic, and then must decide which are preferred to put in the world. The IA puts the most preferable plot points in the world while removing the least preferable points. The Player Modeller's weight similarity calculation, the equation (2) below, is used as a heuristic for determining the preference of points.

$$S_b = W_p \bullet (1 - abs(W_p - W_c)) \quad (2)$$

This 'biased' similarity is calculated from the above equation (2); where S_b is the biased similarity, W_p is the player model weight, and W_c is the compared weight. The result is then subtracted from the maximum similarity to give a final distance heuristic. The similarity value is weighted, or biased, so that playstyles with a greater value will have more influence over the result. Weights have values between zero and one, but this biased similarity calculation will have a maximum result between these values, where the compared weights are equal to the player weights. The final similarity value is calculated by subtracting the biased similarity of a point's weights to the PM weights, from the maximum similarity value.

Using this as a heuristic, the DM decides which points not currently present in the environment, called the 'void-set', should replace points in the environment, called the 'world-set'. Points have potential 'spaces' they could be accessed from. During replacement, each space keeps a list of points trying to replace its current contents, called a competition list. When a point is replaced it is removed from the world set and added to the void-set where it can replace a less preferable world-set point.

Points are given a viability index based on their logical and PM preference. Points with a greater index can replace those with a lesser index.

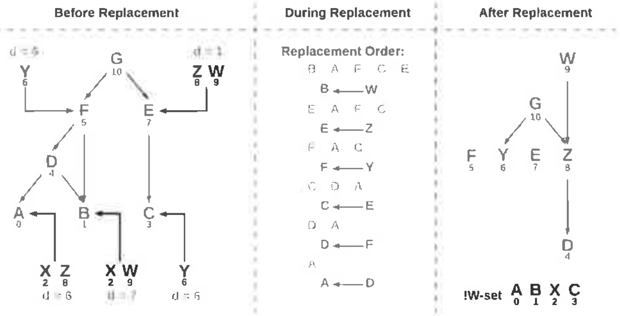

Figure 3: Example of a Round of Replacement

Figure 3 above shows an example of a round of replacement. The left tree represents the points in the world-set and how they could replace other points if they are themselves replaced. The middle column shows the order of replacement and steps through it. This order is determined by the difference in viability index between a space's top competitors. Once replacement is finished the tree becomes that shown on the right.

The values under the letters represent the viability index of competing points. The green values shown in the left tree represent the difference in viability index between the top two competing points. If only one competitor exists, the difference value is equal to the index of the lone competitor.

After each step of replacement, shown in the middle column, the competition lists are updated and new differences in viability indices are calculated. The replacement order is updated and the element space with the greatest difference value is replaced by its best competing point. This process is repeated until no more replacement can be done.

RESULTS

A possibility space containing 150 plot points was constructed, each with their own preconditions and effects, annotated weights, narrative text, and the actions/locations where the point can potentially be found. The dataset allowed for testing a variety of plot structures, with different constraints, so that the most effective design techniques can be found. Created branches led to 6 defined endpoints. Some branches were very specific in their player direction and possible locations, whereas others require more exploration, creating longer paths to the endpoint. 'Difference values' between the PM weights and the weights of the fired plot point was recorded throughout different playthroughs with different endpoints.

It was found that paths with more direction would have smaller PM difference values than those encouraging exploration. The explorative paths were typically longer and made progress towards more different endpoints than the directed paths, and consecutively fired points in the same branch less often than directed paths. Figure 4 below shows how the average PM difference value changes increases with path length. Points with more available locations-to-be-found-in were harder to find compared to those with fewer, but were more likely to be eventually found during longer paths, as the more location-specific points were more likely to be discarded by the DM.

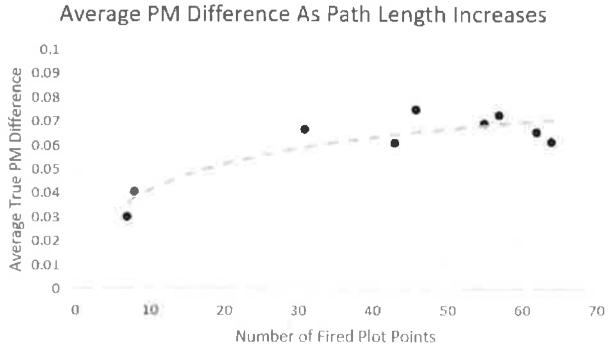

Figure 4: The Average PM Difference Value Increases with Path Length

DISCUSSION

It was determined that possibility space design has a large influence over the effectiveness of the system. Poorly-designed data will lead to poor performance. Exploring multiple separate branches granted improved player agency but made the overall story structure feel less focussed. It is suggested that games adopting this framework implement additional design constraints, simplify the location-specificity of plot points, and provide appropriate direction.

From the researcher's extensive playtesting of the system and its possibility space, this implementation was found to be effective at invoking player agency within well-constructed branches in real-time, yet improvements are needed to ready it for a commercial environment. Much time is needed to implement, debug and tweak the IA, but once the framework is created, it could speed up development of future similar projects. DM systems can also be combined with other AI techniques for specific requirements.

CONCLUSION

This style of Drama Management System offers an excellent supplement to conventional narrative structures, when well-designed and well-constrained, allowing for a great improvement to the variety of discourses available, and to player agency. However, limitations should be placed on the influence of this system over the overall narrative but can still be used to greatly improve the agency of a narrative's discourse. These solutions are also greatly scalable when the possibility space and narrative design are appropriately constrained, and small concessions are made in favour of speed over optimality.

REFERENCES

Hernandez, S.P.; V. Bulitko and E.S. Hilaire. 2014. "Emotion-Based Interactive Storytelling with Artificial Intelligence". *AIIDE.*

O'Neill, B. and M. Riedl. 2011. "Toward a Computational Framework of Suspense and Dramatic Arc" *Affective Computing and Intelligent Interaction* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 246-255.

Procedural Arts (2005) *Façade* [Video Game]. Procedural Arts.

Riedl, M.O. and V. Bulitko. 2012a. "Interactive Narrative: A Novel Application of Artificial Intelligence for Computer Games", *AAAI.*

Riedl, M.O. and V. Bulitko. 2012b. "Interactive Narrative: An Intelligent Systems Approach" *AI Magazine* pp. 2013.

Roberts, D.; M.O. Riedl and C.L. Isbell. 2011. "Beyond Adversarial: The Case for Game AI as Storytelling".

Sharma, M.; S. Ontanon, M. Mehta and A. Ram. 2010. "Drama Management and Player Modeling for Interactive Fiction Games." *Computational Intelligence,* 26, pp. 183-211.

Weallans, A.; S. Louchart and R. Aylett. 2012. "Distributed Drama Management: Beyond Double Appraisal in Emergent Narrative". Springer, Berlin, Heidelberg, pp. 132.

Weyhrauch, P. and J. Bates. 1997. "Guiding Interactive Drama". Carnegie Mellon University Pittsburgh, PA.

WEB REFERENCES

Martens, C. and R. Cardona-Rivera. 2017. *Procedural Narrative Generation.* Available at: https://www.gdcvault.com/play/1024143/Procedural-Narrative (Accessed: Nov 11, 2018).