A MULTIPLAYER CASE BASED STORY ENGINE

Chris R. Fairclough and Pádraig Cunningham,
ML Group, Computer Science Dept.,
Trinity College Dublin,
Dublin 2, Ireland.
chris.fairclough@cs.tcd.ie, padraig.cunningham@cs.tcd.ie

KEYWORDS
Interactive story, multiplayer, case based planning, believable agents.

ABSTRACT

This paper describes the development of an expert case-based character director system which dynamically generates and controls a story, which is played out in a multiplayer networked game world. The system handles multiple users in a game world and directs the non player characters therein to perform for the users parallel storylines, interweaving character roles in each story. The story is told through a 'narrative of actions' and automatically generated dialogue. Much of the storytelling approach is based on the seminal work of Vladimir Propp, to which is applied the AI case based planning paradigm. Initial analysis of the system is based on a review of the system and its output, but future work will involve developing a more objective format for analysis.

INTRODUCTION

The system described here is based on previous work described in (Fairclough and Cunningham 2002). The original implementation was limited to one player taking control of a hero in a simple scripted hero/villain story structure, with no AI storygeneration capability. The current system includes a story director (SD) system which utilises the case based planning (CBP) paradigm. The system also facilitates multiplayer stories, and a range of different possible story structures are allowed for by the approach described in this paper.

The planning and scheduling of stories, modelled as cases in a case based planner, is the primary activity of the system. The game world is run and updated on a C++ server, and a C++ client connects to the server to control a character in the game world. The game type is a 3D adventure game, where the player can use objects and interact with characters. The introduction of a multiplayer element significantly increases the workload of the story director agent. It must handle the current situations of all the players, dynamically identifying possible story cases and assigning story goals that are relevant to each player.

The structure of the rest of the paper is as follows. The second section, entitled '**Background'**, is an overview of the area of computer mediated storytelling, referencing previous work, and describing the genre of computer game that this project is aiming to facilitate. The third section, '**Design'**, details the overall system design and shows how the game mechanics and story mechanics work. The fourth section, '**AI elements'**, concerns the advanced AI techniques utilised in the dynamic story generation subsystem. In the fifth section, an **analysis** of the system and its operation is made, and the paper ends with a description of **future work** and presentation of **conclusions**.

BACKGROUND

The structural analysis of stories has its earliest example in Aristotle's 'Poetics' (Aristotle), and his basic structural elements can still be recognised in popular stories today.

Previous Work

Propp (Propp 1968) is the progenitor of modern analysis, and his structuralist approach is appropriate for computer mediated plot-based storytelling, as it characterises the story as a closed causal system, with the temporally ordered *character function* as the primary building block of the plot. This is the approach used in this project.

There are many modern approaches to story systemation, notably the OZ project in CMU, exemplified in Joseph Bates' work (Smith and Bates 1989) and other work such as Brenda Laurel's drama management system (Laurel 1991), and the work of John Laird's group (Laird and van Lent 2000). The division of the problem into believable agent research and plot management research has been taken from the Oz project and incorporated into this work. This approach has correlations to Propp's work, as he asserts that a character function is independent of the character that performs it.

Story Structure

Although it is technically limited in scope to folktales, a number of authors have noted Propp's structural analysis's remarkable flexibility in its applicability to popular stories. In cinema, TV, and most computer games, a large number of stories have a recognizable structure, hence the predictable nature of a lot of that material. The hero/villain interplay is enriched with the other five character roles in Propp's analysis, yet it remains a simple form. This framework seems suitable for quest-type stories in computer games.

Propp's work has been developed and augmented, chiefly by Bremond (Bremond 1974), yet it is apparent that though

his morphology has been dissected by such authors as Claude Levi Strauss (Levi-Strauss 1958), it remains an outstanding work in the field and still has not been fully explored for its use in computer mediated stories. This could be due to the structural complexity that emerges in the later chapters of the book, in contrast with its initial apparent simplicity in the introduction, when the four principles are introduced. Grasbon's work (Grasbon and Braun 2001) and Ana Paiva's group (Machado et al 2001) demonstrate that varied approaches can be taken to applying Proppian storytelling, each with a successful, yet very different, product.

AI in Story Generation

The use of AI techniques in story generation has been around since the seventies, and emphasis has recently been placed on goal and planning based agent technologies in NPCs (non player characters) (Cavazza et al 2002), (Rizzo et al 1999). The earliest example of this general approach is found in Meehan's classic (yet non-interactive) 'Talespin' (Meehan 1977), which worked by giving characters goals to accomplish, and allowing them work past obstacles placed in their path, helped or hindered by other characters. The use of AI in an 'omniscient' agent which does not inhabit the game world, which monitors and controls the NPCs, and whose activity is directing a story, was seen in (Sgourous et al 1996), and is found in an increasing number of research projects (Mateas and Stern 2000), (Magerko 2002), (Grasbon and Braun 2001), but not, to the authors' knowledge, in currently available commercial games.

MMORPGS

The massively multiplayer online role playing game (MMORPG) is a very new form of game, and the goal is not simply shooting other players, but interacting in a complex, changing environment with teams of other human players. This is an environment which necessitates an author constantly keeping track of the state of the game, and continuously writing a story that takes into account the wishes of the player community, and the dictates of the game world. This requires complex tools that are related to the structure of stories, and it is this need that is targeted by the system in this paper. Although the SD system is built specifically for our game engine, continuation of this work will be centred on the creation of tools for game author/designers.

DESIGN

System Architecture

The system is designed to allow multiple users to log onto a server and control a character avatar that inhabits the game world. The single player game was extended using Winsock and a client-server architecture. The Server updates the client's view of the game world, taking into account the current location, within the world, of the client's player character. Character states and story progress is updated on the server, and the player interface receives commands, which update the client character, and update messages for this character are generated which are sent to the server.

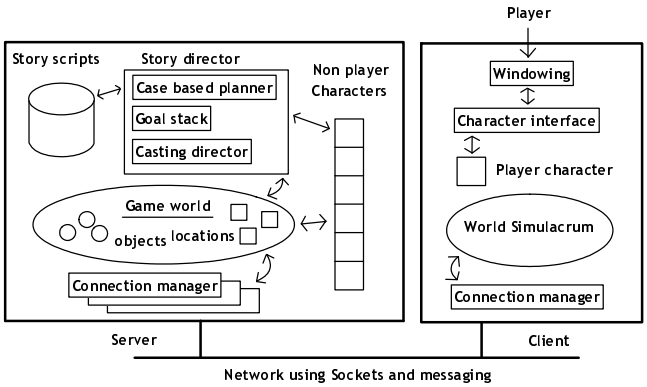

Figure 1: Overall System Architecture

Game Mechanics

The game is based on interactions between characters (NPCs and player characters (PCs)), objects, and locations. The PC has complete freedom of movement within the world, except it may require certain story events to enable transport between locales. A locale is defined as a group of connected locations, similar in concept to the 'freeplay nodes' in the 'StoryNet' of (Swartout et al 2001). The NPCs are modelled in a layered structure, from low level behaviours to higher level targeted goals.

Character Modelling

- Low level - for example, collision detection which steers away from nearby objects and characters as they get too close.

- Social simulation - the NPCs use a basic gossip algorithm and inform other characters of events that have happened to them. They store the order they have met the other characters, and this ranking initially dictates who will be the target of their gossiping.

- Idle behaviours - When an NPC does not have a goal to achieve, they can execute a behaviour such as patrolling around a house or following another NPC. These behaviours are assigned by the story author.

- Targeted behaviour - The SD agent assigns story goals to characters, and the characters search for the object of the goal and execute it. When targeted, a character will not execute idle behaviours.

- Attitudes - characters develop ratings for characters that interact with them, and characters that they hear about via the gossip algorithm. They remember the events that caused these rating changes, so they can gossip about them.

Objects in the game world come in two different types, background objects, and 'action objects' that can be picked up by characters and enable specific actions. For example, a sword enables character A to injure character B which causes a negative change in B's rating for A. Movement around the world can also be achieved with the use of certain 'action objects', for example, a magic carpet. The game mechanics is primarily defined by the use of action objects, and a varied array of different objects has been

defined and each has a number of different uses and effects. A design goal of this project is to allow the progression of the story to be part of the gameplay, while allowing for a variation of gameplay types.

Story Mechanics

The progression of the story must be influenced by the player's movements and actions in the world. To this end, the story director notes character attitudes to the player, which are changed every time a character interacts with the PC, or hears about an interaction with the PC. Characters are each capable of fulfilling a range of story functions.

The story is conveyed through characters performing actions by using action objects and delivering lines to the player. The dialogue is dynamically generated by stringing character names, verbs, and object names together. Witnessed actions are noted and if the player needs to be informed of a certain event, a witness to that event will be assigned the goal of informing them. The player is given choices on how to react to the NPCs. Using polymorphic functions (Grasbon and Braun 2001) is similar to this, and relates to some of Propp's functions (e.g. counteraction - C, reaction - E) where the outcome of the function can be positive or negative, depending on the hero's actions. In example No. 113 in Propp's appendix, there is a repetition of the Donor function, with the first hero reaction function being E-, and the second being E+, after which the hero is provided with the magical object in F+. In this way, Propp unintentionally provided a means for integrating player agency into his model for use in an interactive system.

AI ELEMENTS

The term 'AI' in games usually refers to the character behaviour algorithms, but in this system, the AI elements reside for the most part in the story director agent, and the characters use standard techniques found in many games available today, thus are not addressed in this section. The decision to limit the character AI while localising the AI in the SD is intended to focus the story-centred nature of this work while allowing for its applicability to different character architectures, such as the Proactive Persistent Agent Architecture (MacNamee and Cunningham 2001). This approach is also in line with a new direction in interactive story, outlined in (Szilas 2001), where the importance of making sure the characters behave according to the dictates of the plot, rather than purely according to models of their own motivations, is emphasised.

Expert Knowledge

The expert knowledge represented is that from Propp's work, and consists of a rule based system which works in tandem with the case based system. For each of the 31 character functions that Propp defined, rules are in place that put limits on how that function will be assigned and played out. The character functions enumerated include Villainy, Guidance, Testing of the hero, and Receipt of a magical (useful) object. The Interdiction function, for example, is when the hero receives (I) an order to do something, or (II) a warning not to do something (see Figure 2). This function, like a number of others, is paired with another function, in this case Violation.

Figure 2: An Interdiction Function in the Star Wars Demo Game

The rules consider whether to accept decisions made by role casting, or re-cast a certain story goal. They assign the current goal based on the casting director's roles (see below). They also take into account the user's choices made in feedback(hero reaction) functions. The rule based system effectively draws the decisions made by the other components together, before the characters can act them out. An example using pseudo- code:

If (currentGoal.functiontype = Villainny)
And (currentmove.currentVillain.suitability > threshold)
Assign (currentGoal TO currentmove.currentVillain)

Case-based Planning of Stories

Case based reasoning is a popular AI field which aims to solve problems by extrapolating from past, solved, problem situations to find solutions for new problems, adapting them, as needed, to the dictates of the current situation. The k-nearest neighbour algorithm is used to find cases that are similar to the input case. This approach has been identified as suitable for constructing story scripts from a base of authored scripts.

Each case in the case base represents one *move* of a story (as defined by Propp), and consists of a script that is interpreted and assigned by the SD. A script line can contain very abstract instructions, such as a one word 'Villainy' instruction. This will trigger the SD to finding the character with the villain role, (or a character socially close to the villain) and instruct it to perform an act of villainy on the hero or a character close to the hero, such as a murder. More specific instructions, such as assigning the mediator character the goal of notifying the hero of a villainy event can also be scripted. The cases are converted to a goal stack by the SD to be assigned to the NPCs. Below is an example, from Propp, of a two move story, converted to script form. Each move is represented as a case in the case based planner.

No. 95 in Appendix 3 of (Propp68):

Move I
villainy (expel) ;
mediation (transport) ;
donor (test) ;
reaction ;
provision (wanted) ;
return ;
Move II
lack ;
mediation (transport) ;
counteraction ;
departure ;
donor (test) ;
reaction ;
provision ;
return ;

Move I above requires the following resources, and thus when they are available it would have a good likelihood of being selected: a villain, a mediator, a donor (or multiple candidates), objects that can perform the actions 'expel', 'transport', and 'test', and an object that is 'wanted', e.g. treasure. While carrying out the direction of this script, the SD will assign the goals to specific characters and objects.

There are 80 cases in the case base, from the 44 multi-move story scripts given by Propp in his appendix 3. In the k-nearest neighbour part of the algorithm, the cases are compared to the current state of the story world to find the best fit for a case from which to select the next story goal. The comparison is based on an analysis of the character and object resources needed to execute a story script, and also on past story functions, and how they were performed. Taking the sphere of action of the villain as an example, its contribution to the suitability rating of an individual story case will be proportional to (i) the number of functions involving the villain that exist in the case, (ii) the number of characters that could perform as a villain in the current locale of the PC, and (iii) their suitability level.

In the story that the example above is abstracted from, move II directly follows move I, and the link between cases that follow each other is preserved in the CBP system. Thus, move II would be more likely to be executed after move I has completed than other candidate cases. Cases similar to move II would also be considered. The cases in this system are best described as story templates, whose details are filled in by the SD. This 'filling in' is done by the casting component, as described below.

Casting

A character that is assigned a story goal takes one of nine roles. Propp defined the hero, the villain, the mediator, the donor, the helper, the false hero, and the princess as the seven 'spheres of action', and these are augmented with the roles of the family and the king. The characters in these roles can be initially defined by the story author, yet are dynamically reassigned by the story director agent during the game, according to the relationships of the characters. Casting roles to characters, and finding objects that can help fulfil goals is a task that is modelled by rules that encode the appropriate constraints. The trend of modern successful games is towards large, complex, simulated game worlds containing many objects and characters. In this environment, for a dynamically generated storyline to be consistent, a mechanism must be provided for identifying the appropriate object or character for a given story goal. To find the right tool to achieve a certain goal, the properties of each must be searched to fit the constraints dictated by the current goal. In this game engine, characters and objects have properties which are matched to the dictates of the story function being carried out.

Each story function (of the 31) has a set of constraints for a character to satisfy, and characters have social ratings, possessed objects, loyalties and a current location that are matched as well as possible to the constraints. The character that matches the constraints the best will be assigned the goal. Objects also have different types, and a character can be assigned an intermediate goal to find the right one, in order to perform a certain goal. For example, a sword would be used to injure a character instead of a magic carpet, and if a character needs to acquire the object, a goal to pick it up or be given it is added to the goal stack for that move.

The example script from the previous section would be played out after it has been selected. The character direction would adapt to any changes in the state of the game world. For example, in the Donor test function in move II, the connected provision function serves as a liquidation of the initial lack, as there is no explicit liquidation function. That particular lacked object would change hands during the course of the story until it coincides with a donor character.

Multiple Parallel Storylines

For a multiplayer environment, the SD handles the story from multiple viewpoints. The main SD instantiates a new story director for each player, each looking after most of the planning and casting relevant to each. Two players could both act as hero characters for two separate hero stories using intersecting sets of NPCs. In this case, there are two SDs maintaining a set of roles and list of story functions for each player's hero story. Alternatively, one player could be the villain or the false hero in the same story as the first player's hero. Ideally the SD could dynamically recognise the roles that each player wishes to adopt, based on their playing, but a set of options could also be presented to the player as they join the game world, to find out what sort of character they want to portray. The latter approach is being developed for use in an evaluation version of the multiplayer game.

Even in the single player game, it has been found that multiple parallel storylines need to be executed, due to the nature of a non-linear, free-roaming type of game such as ours. A player may leave the locale of a story case being executed, to enter a locale or situation where a different case would be more suitable. To account for this, the SD can plan and assign a number of different story cases at once per player.

ANALYSIS

The primary task in analysing the usefulness of this work consists of evaluating how the quality of the stories generated by it is maintained, while allowing interactivity. The structures described by Propp are derived from his study of folktales, and the stories generated by this system are generated from those structures. The validity of the whole approach relies on Propp's assertion that these structural features are the primary classifying features of the stories he analysed. This being true, the structures should form a good basis for the generation of new stories. Propp was careful in his work not to claim generality of his morphology, yet even so, an engine that produces a specific genre of story in adaptive variations is still desirable. To enable more generality, a more complete narratology could be implemented, including the work of Bremond, Greimas, and Barthes, among others.

An emphasis in both Propp and Aristotle's work is that of the conception of the story as whole as a causal system. Ideally, every event that happens in the story world has either an effect on some other story event, or is an effect of one. The system described here allows for this by abstracting the social, and action-object based game events from their context, effecting attitudes each character has for the others. These ratings, along with the choices made concerning preceding story events, are fed back into the SD agent's deliberations on generating the next story goal.

Completed Demo Games

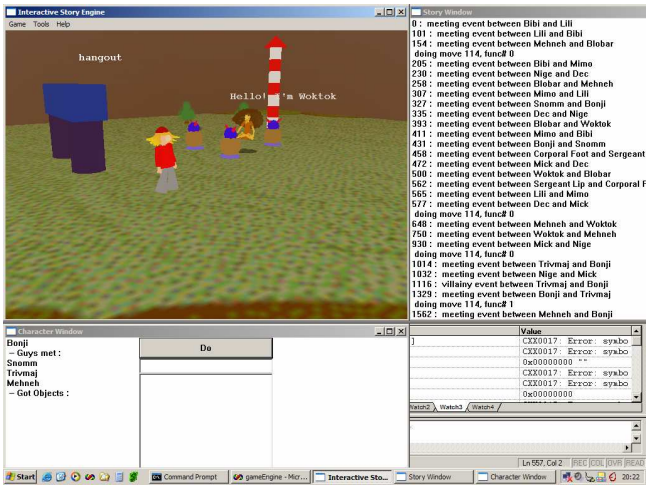

Figure 3: 'Bonji's adventures in Calabria' with Story Progress Window and Character Status & Control Window

Three demo games have been developed to date, the first a simple original environment entitled 'Bonji and the magic peanut'. The other is based on the first half (move?) of 'Star Wars – A new hope', TM Lucasfilm, and features the familiar characters in a more variable form of the traditional 'rescue the princess' plot. The characters are allowed some autonomy and Luke is channelled through the story not by limiting his location, but by giving story goals to nearby characters. The third demo game incorporates the multiplayer option, as well as a wider range of abilities in the player interface, facilitated by a more interesting set of action objects. The case based SD system is fully integrated with this game, 'Bonji's Adventures in Calabria' (Figure 3).

Advantages

There are three main advantages of using a system such as this in a game engine. Firstly, the ability to choose different paths through the same story is a powerful means to increase replay value of a game. This is enabled by allowing different characters to achieve the same story goal in their own ways.

Secondly, allowing the player to influence the progress of the story is enabled by the use of a SD agent. Different story structures are brought into the game according to player choices and the world state using case based story planning. The range of cases available can be authored for each game world, allowing the story author a higher level control of story events.

Thirdly, with a multiplayer option, and because the story generation is real-time, players can interact in more complex ways, playing with NPCs' loyalties to gain the upper hand over each other. This will allow for adventure-game type stories that remain consistent as players meet other players. A common element in RPGs is a team of characters under player control, and in multiplayer versions, players can meet up with others to form a team. Many online MMORPG developers have had problems in implementing a satisfying storyline in this type of environment, as each player may take different paths through the game, teaming up when they see fit.

FUTURE WORK

Two future additions to the story engine are:

Learning – In case based systems, the 'retain' step consists of storing the new solution that has been adapted from the old case, or the combination of multiple old cases. This facilitates learning from experience, and for it to work in the story engine, an algorithm for properly (according to certain rules) combining two or more cases is needed. The SD would then be able to generate and learn new cases and find good fits for new situations more easily.

Deception – When a NPC uses deception, they inform the player of events that did not happen in the game world. The fabricated events should be calculated to induce a certain reaction in the player. For instance, to make the player think NPC1 is on the side of the villain, NPC2 could tell the player that NPC1 did an act of villainy on a character that is close to the player. Propp's role of 'false hero' has close links with deception and relates to the hero uncovering the deception of one who takes responsibility for the acts of the hero. However, deception could also be used in other parts of the tale. Deception was used as a dramatic focus in the 'Brutus' system (Bringsjord and Ferucci 1999).

The game engine, as is, is not scaleable for use in a large scale modern game world, but an important aim of the project was to make the architecture scaleable. To facilitate the creation of tools for the incorporation of the story director system into other games, it has thus been designed

with the complexity of a large-scale simulated world in mind.

CONCLUSIONS

Currently, analysis of the system is based on a review of the system and its output. In the interest of a more objective analysis, the networked system will be combined with a separate user interface for getting feedback on the users' experiences with the game, based on story criticism criteria. The result of this will include an analysis of believability, consistency, drama, and the level of user interactivity.

As Young notes in (Young 2000), character and plot have a symbiotic relationship, and while this system relies heavily on a plot based model of story, characterisation is seen as an aspect of interactive storytelling that relies heavily on the designer/author of the story world. The characters in the story, although they can all be assigned different roles by the SD, have behaviours that are not so assigned, and it is in these that an author can bring out individual characteristics. The important thing is to make sure that how a character performs a story function is consistent with its characteristics.

The system described here is unique in its combination of AI techniques, software architecture, and game style. However, there are other systems which take into account emotional modelling, cinematography, and others of the multitude of elements which make up compelling storytelling. Magerko's work (Magerko 2002), among others, is similar in approach to ours, and this is encouraging as it seems to be a useful approach to the problem. There are other, increasingly varied approaches to computer mediated storytelling and storygeneration, and this is an indication of the richness of the subject matter. Stories can be told in many ways and human imagination will always be the best way to create them, but systems such as these can provide advanced tools for the creation of the next generation of story vehicles.

REFERENCES

Aristotle. *Poetics.*

Bremond, C. 1974 *Logique du Recit*. Seuil.

Bringsjord, S. and Ferucci, D. 1999. *Artificial Intelligence and Literary Creativity: Inside the Mind of Brutus, A Storytelling Machine*. Laurence Erlbaum, Mahwah, NJ.

Cavazza, M., Charles, F., and Mead, S J. 2002. "Character-Based Interactive Storytelling" IEEE Intelligent Systems Vol 17-4 pp 17-24

Fairclough, C. and Cunningham, P. 2002. "An Interactive Story Engine", AICS 2002, LNAI 2464. Springer-Verlang. pp 171-176.

Grasbon, D. and Braun, N. 2001. "A Morphological Approach to Interactive Storytelling" *Proceedings of on Artificial Intelligence and Interactive Entertainment. Cast '01*, Sankt Augustin Germany

Laird, J. and van Lent, M. 2000. "Human level AI's killer application - computer games" AAAI National Conference on Artificial Intelligence.

Laurel, B. 1991. *Computers as Theatre*. Addison-Wesley.

Machado, I., Paiva, A., Brna, P., 2001. "Real Characters in Virtual Stories – Promoting Interactive Story Creation Activities" LNCS 2197 pp. 127 – 134.

MacNamee, B. and Cunningham, P. 2001. "A Proposal for an Agent Architecture for Proactive Persistent Non Player Characters", *Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science*, pp. 221 - 232, 2001.

Magerko, B. 2002. "A Proposal for an Interactive Drama Architecture", *AAAI Spring Symposium on interactive entertainment.*

Mateas, M. and Stern, A. 2000. "Towards Integrating Plot and Character for Interactive Drama." *Working notes of the Social Intelligent Agents: The Human in the Loop Symposium*. AAAI Fall Symposium Series.

Meehan, J. 1977. *The Metanovel: writing stories on computer.* University microfilms international.

Propp, V. 1968. *Morphology of the Folktale*. University of Texas Press.

Reilly, W. S. and Bates, J. 1992. "Building Emotional Agents" Technical Report CMU-CS-92-143.

Rizzo, P., Veloso, M.V., Miceli, M. and Cesta, A. 1999 "Goal-based Personalities and Social Behaviours in Believable Agents." *Applied Artificial Intelligence*, 13:239–272.

Sgouros, N.M., Papakonstantinou, G. and Tsanakas, P., 1996. "A Framework for Plot Control in Interactive Story Systems", Proceedings AAAI'96, Portland, AAAI Press.

Smith, S. and Bates, J. 1989. "Towards a Theory of Narrative for Interactive Fiction", Technical Report CMU-CS-89-121

Levi-Strauss, C. 1958 *Anthropologie Structurale*. Plon.

Swartout, W., Hill, R., Gratch, J., Johnson, W.L., Kyriakakis, C., LaBore, C., Lindheim, R., Marsella, S., Miraglia, D., Moore, B., Morie, J., Rickel, J., Thiebaux, M., Tuch, L., Whitney, R. and Douglas, J., 2001. "Toward the Holodeck: Integrating Graphics, Sound, Character and Story". *Proceedings of the Autonomous Agents 2001 Conference*, Montreal, Canada.

Szilas, N. 2001. "A New Approach to Interactive Drama: From Intelligent Characters to an Intelligent Virtual Narrator". Proceedings of Spring Symposium on Narrative Intelligence, AAAI press. Pp 72-76.

Young, R.M., 2000. "Creating Interactive Narrative Structures: The Potential for AI Approacches". *AAAI Spring Symposium in Artificial Intelligence and Entertainment*, AAAI Press.