A Framework for Plot Control in Interactive Story Systems

N. M. Sgouros, G. Papakonstantinou, P. Tsanakas

Department of Electrical & Computer Engineering
National Technical University of Athens
Zographou Campus, Greece, 157 73
sgouros@dsclab.ece.ntua.gr

Abstract

This paper presents a framework for plot control in interactive story systems. In this framework, the user takes the place of the main character of the story, the protagonist. The rest of the cast consists of discrete characters, each playing a specific role in the story. A separate module in this system, the plot manager, controls the behavior of the cast and specifies what the protagonist can do. The story plot is dynamically shaped by the interference between cast members and their social interactions. The system accepts as input a story map which provides the main metaphor for organizing the plot and localizes the interaction of the protagonist with the rest of the cast. We are implementing this framework in PEGASUS, an interactive travel story environment for Greek mythology.

Introduction

Research in interactive story systems aims to create a new computer-based art form, providing experiences that are both meaningfully interactive and good stories (Waters 1995). Current interactive story systems can be classified into two categories: (i) story graphs and (ii) simulated worlds. In a story graph the user follows links between predefined episodes. In a simulated world, the user interacts with computer-simulated characters in a virtual environment. Unfortunately, story graphs are only minimally interactive, while, in the majority of cases, the interaction with a simulated world is not coherent enough and it does not have a temporal structure that could classify it as a story. Therefore, research in plot control is crucial for further development of interactive story systems.

Travel narratives are stories in which the main character progresses through a series of episodes or encounters, until it reaches a final destination. Some of the oldest and most popular story forms can be classified as travel stories (e.g. the Odyssey, Alice in Wonderland, etc.). The fact that a lot of travel stories have weakly constrained plots allows their presentation in highly interactive forms (e.g. rhapsodies in ancient Greece). Consequently, we believe that the richness of material that travelogues can encompass, the nature of their plot, and the wealth of characters they can support, makes them ideal for the creation of computer-based interactive stories.

This paper presents a framework for plot control in interactive story systems. Figure 1 shows the framework structure. The main module in this system is the plot manager (PM). PM shapes the protagonist's interaction with the story system by controlling what the cast members do and specifying what the user can do. PM consists of a set of rules for social action, the specifications for each character role in the story, and a user interface manager. The story plot is dynamically shaped by the interference between cast members and their social interactions. Cast members interfere with each other, either by managing resources that influence the service of their goals, or by reacting to social norms or values relevant to their actions.

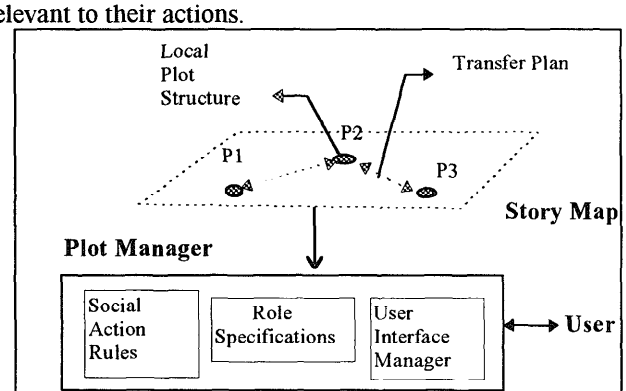

Figure 1: Architecture of the interactive story environment.

PM accepts as input a story map consisting of a set of points (P1, P2 and P3 in the figure). This map provides the main metaphor for understanding the narrative. Furthermore, it serves as the organization for storing plot control and background knowledge structures. More specifically, each location in the map has a *plot structure* associated with it. This structure contains the local cast members, their roles and relations, and it is activated when the user reaches this location.

Furthermore, pairs of points in the map have *transfer plans* associated with them. These describe the conditions under which the user can move between these points. In

addition, they specify the resources required for the execution of the transfer plan and ways for acquiring them. At each point in the story, PM communicates with the user and coordinates the interaction between its different subsystems.

This framework strikes a balance between opposing features such as interactivity and plot control. In particular, it allows the user to have a lot of control over what happens, by taking the place of the main character of the story and deciding on its current set of goals and actions. On the other hand, it controls the behavior of the cast using constraints derived from their role specifications and a set of social action rules that govern their social interactions.

The rest of this paper is organized as follows. Section 2 describes the methods by which character interference takes place in the story environment, along with the specifications for the roles played by the cast. Section 3 describes the rules for social action in the PM. Section 4 describes the user interface. Section 5 presents the algorithm for the plot manager, while section 6 presents an example of a story created in PEGASUS and gives some details on the implementation of this environment. Finally, section 7 presents related work, while section 8 is a conclusions and future work section.

Character Interference & Roles

There are two types of plot interference between cast members; *goal* and *normative*. In goal interference a character X tries to influence the service of the goals of some other cast member Y in any of the following ways:

- *Favorable Interference*. X helps Y through the acquisition or use of exceptional resources or plans that may serve Y's goals. Examples in PEGASUS include X using weapons endowed with special powers (e.g. the shield of Athena) on the side of Y during a battle, or X helping Y win the support of Poseidon, the sea god, if Y is to embark on a sea journey.
- *Unfavorable Interference*. In order to make life difficult for Y, X introduces two kinds of problems affecting Y's resources and social relations.
  1. Loss. Possible choices include theft, breakdown or physical destruction of resources (e.g. fire).
  2. Shortages. X creates shortages over necessary resources for Y. These may range from forbidding their use, if X plays the king role, to forcing Y to share these resources with other cast members.

In normative interference, a character X seeks to influence positively or negatively the compliance of Y to a set of social norms or values. In the case of positive influence, X checks whether Y's behavior violates any moral, or legitimate rules. If any transgression is observed, Y is notified and proper punishment is planned by X. In the case of negative influence, X tries through ingratiation or reciprocity to lure Y away from compliance.

The story map assigns a set of thematic roles to each cast member (except for the user). Each character pursues only types of interference consistent with its role. Currently we have developed seven thematic roles.

- The *Judge*. This role seeks to enforce legitimate power over the story characters. It checks for transgressions of character behavior from the law and enforces penalties in cases of deviation. The judge pursues positive normative interference.
- The *King* is concerned with political aspects of the behavior of story characters. It checks for political repercussions of each character's actions. These can include conflicts over public resources, disagreement over authority rulings or public policies. In these cases the king is the voice of the political establishment and tries to maintain the status quo. Furthermore, the king can issue new orders with which the rest of the cast should comply. Both goal and normative interference are consistent with this role.
- The *Rebel*. The purpose of this role is to oppose the actions of the political establishment. It disagrees with what the king proposes and uses unfavorable interference strategies to impede the king's goals. This may bring him in opposition with the judge and the priest as well. The rebel pursues negative normative interference.
- The *Priest*. This role has the goal of making the characters comply with religious norms. The priest suggests to the protagonist plans consistent with religious beliefs and practices. Furthermore, it informs the user of divine attitudes regarding its actions. Finally, the priest evaluates character behavior in terms of religious beliefs. This role pursues positive normative interference.
- The *Messenger*. The goal of this role is to inform the user on off-stage developments. PM uses these character types to present to the user a coherent and entertaining account of the current story context.
- The *Sage*. This role offers expert advice in aid of the protagonist. PM supports different instantiations of this role based on the current story context. Thus in a battle scene the sage can be an oracle which advises the user on the battle outcome, or in the case of a sea journey it can be an experienced sailor. Only favorable goal interference for the protagonist are consistent with this role.
- The *Villain*. This role represents all things evil in the story. The villain checks for profit opportunities independent of the legitimacy or the morality of its actions. The villain practices negative normative and unfavorable goal interference.

In addition to these role features, PM also contains a set of conflict management rules. These are domain-specific rules that specify a set of conditions under which a character playing a given role prevails over its opponents. For example, in the case of Greek mythology, one such rule says that cast members supported by at least one god, prevail over opponents with no divine support.

Social Action

The behavior of each particular character is a combination of the types of interference associated with its role and its interaction with the rest of the cast. This interaction is shaped by the application of a series of rules governing social action in the story environment and described below.

Cooperation

The cooperation rule states that whenever two characters X and Y mutually believe they have a common goal G and complementary resources for it, then they pursue favorable goal interference between them with respect to G.

Reciprocation

The positive reciprocation rule states that character X seeks to favorably interfere with some of character Y's goals relativized to the belief that Y has or will adopt some of X's goals. The negative reciprocation rule describes states of mutual aggression in which X seeks to unfavorably interfere with some of Y's goals, believing that Y has done the same thing.

Cooperation and positive reciprocation leads to the formation of groups of allies in the story, i.e. characters tied with common goals or positive reciprocity commitments. PM assigns a leader to every group of allies. This is the cast member with the largest number of common goals or reciprocation commitments with the rest of the group (i.e. the one who cares most about the group). In addition, the group has an agenda consisting of the common goals of the group members. The leader assigns tasks to group members based on this agenda. Group members have to follow their task assignments and favorably interfere with the service of each goal in it.

Group Performance

Each character X monitors the progress of the goal agenda of the group it belongs relative to its own goals. Furthermore, X monitors the compliance of the rest of the group with this agenda. If X discovers that its goals cannot be served by the current group, then X defects from its group and seeks new allies. X also defects, if it discovers that any of the other group members (e.g. Y) unfavorably interferes with any of the goals in the agenda. In the later case, X and Y become enemies through the application of the negative reciprocation rule.

Exchange

The exchange rule states that character X can use the resources supplied by some other cast member Y, if he offers some other resource or service specified by Y. The exchange rule governs the way characters trade the resources at their disposal. X can override this rule through theft or deception. In both cases, X and Y become enemies according to the negative reciprocation rule.

User Interface

The goal of the user interface is to specify what the user can do, as the story unfolds. The user interface manager consists of two subsystems: the action menu and the storyboard.

Action Menu

At each point in the story, PM composes dynamically a menu of possible user actions based on the protagonist's actions and the behavior of the rest of the cast. These actions include possible reactions to interference from other cast members, character-specific actions for interacting directly with a cast member and resource-specific actions for following transfer plans.

User reactions. User reactions are consistent with the character behavior rules described in the previous section. In the case of favorable interference from another character X, PM applies the positive reciprocation rule and proposes to reward X. If the protagonist chooses this option, possible actions include joining any of the groups led by X, or providing X with resources or services necessary for accomplishing its goals. Choosing the later leads the user to a set of character-specific actions determined by the thematic roles played by X and its current goals.

In the case of unfavorable interference from X against the protagonist, PM presents two options to the protagonist:

- Replenish resources using the exchange rule.
- Retaliate against X based on the negative reciprocation rule.

The user has the option of replenishing its resources in case of loss. Choosing this option leads the user to a set of possible trade agreements with characters that have the necessary resources. If the user decides to retaliate against X, it can do it either directly, by confronting X, or by joining groups that oppose X's allies.

In the case of normative interference from X, the user can choose either to obey the constraints introduced by X or to transgress the norm.

Character-specific actions. PM composes a set of actions specializing the decisions made by the protagonist using the reactions of the previous section. These actions are geared towards a specific character X. Composition takes into account the thematic role played by X, the groups X belongs to, its current goals and a set of plans stored in PM's memory for achieving these goals. For example in PEGASUS, if the protagonist asks the help of a priest to appease some god, then this priest will suggest a sacrifice or a sponde to this god. Both actions are role-specific to the priest.

Resource-specific actions. Each time the protagonist follows a particular transfer plan in the story, PM composes a set of buttons prompting the user to acquire all the resources specified in the plan.

Storyboard

The storyboard has three main goals:

- Provide the user with a metaphor for understanding the development of the story.
- Give to the user concise directions for navigating in the story environment.
- Dramatize character behavior.

The interface satisfies all these requirements by diving the screen space into distinct functional areas (Figure 2). The first area (upper left of Fig. 2) depicts the story map, which in PEGASUS coincides with a geographical map of the Aegean.. This graphical description indicates the current location of the protagonist, its final destination, the places he has already visited and, depending on the current context, possible intermediate destinations. Whenever the user decides to move between places, the user interface instantiates the transfer plans connecting these points. In PEGASUS, for example, the map contains a Take-Sea-Journey-Plan with the resources shown in Figure 3c for traveling between mount Pelion and Troy, two places lying on opposite coasts in the Aegean.

The second area (upper right of Fig.2) displays the action menu composed by PM. This is a button-based interface that provides the user with alternatives for interacting with the rest of the cast.

Finally, PM uses the rest of the screen for dramatizing character behavior using a set of multimedia techniques (e.g. animation, still images, text clips etc.).

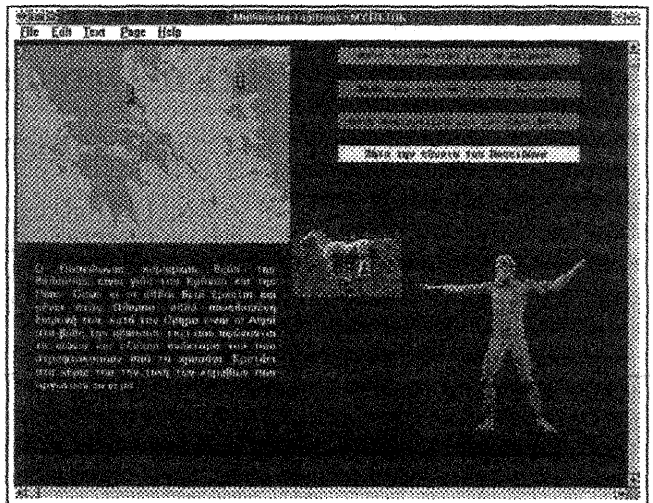

Figure 2: Screen design for the PEGASUS interactive story environment.

The Plot Manager Algorithm

PM is controlled by an event-driven algorithm, with the story map as the focus of activity. In particular, when the user chooses to move to a point in the map, PM executes the following commands:

1. Instantiate the transfer plan connecting the two points.
2. Instantiate the local plot structure at the destination point.
3. Run role and social action rules.
4. Update the user interface.

When the user reaches its current destination, PM executes the following sequence in a loop:

1. Update the user interface.
2. Run role and social action rules.
3. Run conflict management rules.

The loop terminates when the user decides to move to another point in the map.

An Example

In the beginning, PEGASUS displays a text clip that sets the context for the current story. According to this introduction, the gods warn Cheron, the wisest of the Centaurs, that Achilles, his favorite grandson, will die in the battle of Troy. Worried about the fate of the famous hero, he asks the user to help him make the journey from its home, in mount Pelion, to Troy, to protect Achilles.

When the user accepts this invitation the following things happen. PM instantiates the plot structure for Pelion (see Figure 3a). This plot structure suggests that Cheron is the only cast member active in Pelion. Furthermore, it suggests that Cheron is an ally of Achilles and that the protagonist and Cheron have a common goal

to protect Achilles. These goal statements satisfy the cooperation rule, therefore PM creates the first group in the story with the members and the agenda shown in Figure 3b. Finally, a button showing Troy on the map appears and the transfer plan from Pelion to Troy is instantiated. This plan suggest that our heroes should travel by boat. PM composes and presents a text clip on screen in which Cheron, in its sage role, suggests to the user a sea journey. In addition, PM creates a button for this option in the interface.

```
(defPlotStructure Pelion                                    (a)
  Cast ((Sage Cheron))
  CastRelations ((Allies Cheron Achilles)
                 (Goal User (Protect Achilles))
                 (Goal Cheron (Protect Achilles))))
-------------------------------------------------------------------
(Group #:group1 (Cheron User))                              (b)
(Goal #:group1 (Protect Achilles))
-------------------------------------------------------------------
(defPlanResources (Take-Sea-Journey ?start ?dest)           (c)
    ((Resource Ship (Take-Sea-Journey ?start ?dest))
     (Resource Crew (Take-Sea-Journey ?start ?dest))
     (Resource (Supply Food) (Take-Sea-Journey ?start ?dest))
     (Resource (Supply Water) (Take-Sea-Journey ?start ?dest))
     (Resource (Support Poseidon) (Take-Sea-Journey ?start ?dest))))
-------------------------------------------------------------------
(defPlotStructure Corinth                                   (d)
  Cast ((Priest Anacleon Poseidon Corinth)
        (King Eumeneas Corinth)
        (King Agamemnon Mycinae)
        (Sage Cheron))
  CastRelations ((Allies Eumeneas Agamemnon)
                 (Allies Cheron Achilles)
                 (Enemies Achilles Agamemnon)))
```

Figure 3: Plot elements in PEGASUS.

When the user clicks on this option PM instantiates the Take-Sea-Journey plan and activates the structure representing its prerequisites (Fig. 3c). Based on this structure, PM composes an action menu that prompts the protagonist to acquire all these resources. One of the prerequisites for this plan is that Poseidon, the sea god, should bless this journey. When the user clicks the button corresponding to this option, the goal of worshipping Poseidon is pushed onto the agenda for #:group1. Furthermore, PM composes a text clip in which Cheron, in his sage role, suggests to the user possible Poseidon temples. The location of these temples automatically appear as buttons on the map.

Let us assume that user clicks on Corinth, a big port near Athens with one of the biggest Poseidon temples. Then PM loads the plot structure associated with it (Fig. 3d) and an icon on the map starts moving from Pelion to Corinth. This plot structure suggests that there are four cast members active in Corinth, Anacleon the local priest for Poseidon, Eumeneas the king of Corinth, Agamemnon the king of Mycenae and Cheron as the sage. Furthermore, it suggests that Eumeneas and Agamemnon as well as Cheron and Achilles are allies, while Achilles and Agamemnon are enemies. These relations and #group1's goal which is to protect Achilles, an enemy of Agamemnon, trigger the negative reciprocation rule. Based on this rule, Eumeneas opposes #:group1 and seeks to unfavorably interfere with its goals. Eumeneas notices that a shortage type of interference is possible and, as a king, issues an order forbidding Cheron and the user to worship Poseidon in Corinth.

The priest on the other hand, interprets the royal order as a transgression from the religious belief that everyone has the right to worship the gods. As a result, Anacleon pursues positive normative interference and seeks to oppose this order. PM uses its conflict management rules and decides that Anacleon will prevail in this conflict because he is supported by Poseidon, while Eumeneas receives no divine support. Then it instantiates the Wrath-of-Poseidon dramatic script which involves sound effects of a sea storm. Finally, it informs the user that Poseidon had its way and now Eumeneas allows our heroes to worship the sea god. The story continues and our heroes try to obtain all the resources specified in the Take-Sea-Journey plan.

PEGASUS runs on a Windows PC. The plot manager is built on top of an ATMS-based rule engine written in C++, similar to the one described in (Forbus & de Kleer 1993). The multimedia interface was built using Toolbook, a multimedia authoring system for the PC. We have implemented a Dynamic Link Library (DLL) in Windows that allows PM to communicate with Toolbook. Currently, we have implemented the user interface manager, the cooperation and reciprocation rules and the specifications for the king and the priest roles in the PM. We work on implementing the rest of the framework.

Related Work

Recently, there has been considerable research in the creation of believable interactive characters. This work has concentrated mainly on portraying the emotional state of these characters (Bates 1994), on supporting full-body interactive video environments (Maes 1995), or on developing directed improvisation paradigms in which computer characters improvise a joint course of behavior following users' directions (Hayes-Roth & Brownston & Sincoff 1995). Our research complements this work, focusing on effective plot control and higher level social interactions between story characters.

Our model for the social behavior of story characters views social action as a complex phenomenon emerging from the interaction between cognitive agents (i.e. agents endowed with cognitive representations of goals and the capacity to pursue them). This is a view shared by many social and cognitive psychology researchers (Castelfranchi & Conte 1995, Hogg & Vaughan 1995, Ortony et al.

1988) and forms the basis for constructing the social action rules in this framework .

Map or labyrinths have been used as the primary metaphor for organizing stories in hypertext, MUD or MOO systems (Barrett & Redmond 1995). We have adopted this metaphor in our work and showed how it can be used for interactive story generation.

Storytelling systems have been constructed for use in educational applications (Schank & Fano. 1992). These programs support sophisticated indexing mechanisms that allow the user to find and access stories relevant to its needs, from a large database. Our work provides a framework that can enrich the storytelling capabilities of these systems by allowing for interactive renditions of the stories they deliver.

Conclusions & Further Work

We have described a framework for plot control in interactive story systems. In this framework, the story plot is dynamically shaped by the interference between cast members and their social interactions. A separate module in this system, the plot manager, controls the behavior of the cast using a set of specifications for the roles played by the cast members and a set of rules for social action. Furthermore, the plot manager specifies what the user can do via the user interface. As a result, the system is meaningfully interactive while achieving adequate plot control.

We are extending this framework by developing higher level plot control techniques that will preserve thematic coherence and prevent the role and social action rules from overgeneration. These techniques consist of local character behavior filters that restrict the set of possible actions of the cast to those that are 'relevant' to the current plot. Moreover, we are developing more complex social action rules that describe concepts such as altruism or love and we are extending the repertoire of character roles to enrich the character types supported in this environment. Finally, we are exploring ways of incorporating plot elements from traditional story environments in our interactive system, based on the work cited in (Lehnert 1981, Lakoff 1972).

This work is part of a bigger project for developing the next generation of electronic books. We are cooperating with pedagogical institutions in the University of Athens to integrate our framework into a series of activities that foster creativity and enhance the enjoyment of standard reading material.

Acknowledgements

This work has been partially funded by the Greek Secretariat for Research & Technology (GSRT).

References

Waters, R. C. 1995. The 1995 AAAI Spring Symposia Reports; Interactive Story Systems: Plot and Character, *AI Magazine*, 16(3): 8-9.

Barrett, E., Redmond, M. eds. 1995. *Contextual Media.* Cambridge, Mass.: MIT Press.

Bates, J. 1994. The Role of Emotion in Believable Agents, *Communications of the ACM,* 37(7): 122-125.

Conte, R., and Castelfranchi, C. 1995. *Cognitive and Social Action.* London: UCL Press.

Forbus, K. D., and de Kleer, J. 1993. *Building Problem Solvers.* Cambridge Mass.: MIT Press.

Hayes-Roth, B., Brownston, L., Sincoff, E. 1995. Directed Improvisation by Computer Characters, Technical Report, KSL-95-04, Dept. of Computer Science, Stanford Univ.

Lakoff, G. 1972. Structural Complexity in Fairy Tales, *The Study of Man* 1: 128-150.

Lehnert, W. 1981. Plot Units and Narrative Summarization, *Cognitive Science*, 4:293-331..

Maes, P. 1995. Artificial Life meets Entertainment: Lifelike Autonomous Agents, *Communications of the ACM*, vol. 38(11).

Hogg, M. A., and Vaughan, G. M. 1995. *Social Psychology: An Introduction.* London: Prentice Hall.

Schank R., and Fano A. 1992. A Thematic Hierarchy for Indexing Stories in Social Domains, Technical Report, # 29, The Institute for the Learning Sciences, Northwestern Univ.

Ortony A., Glore G. L., Collins A. 1988. *The cognitive structure of emotions.* New York: Cambridge University Press.