Evaluating Directorial Control in a Character-Centric Interactive Narrative Framework

Mei Si, Stacy C. Marsella and David V. Pynadath
Institute for Creative Technologies
University of Southern California
13274 Fiji Way, Marina del Rey, CA 90292
{meisi,marsella,pynadath}@ict.usc.edu

ABSTRACT

Interactive narrative allows the user to play a role in a story and interact with other characters controlled by the system. Directorial control is a procedure for dynamically tuning the interaction towards the author's desired effects. Most existing approaches for directorial control are built within plot-centric frameworks for interactive narrative and do not have a systematic way to ensure that the characters are always well-motivated during the interaction. Thespian is a character-centric framework for interactive narrative. In our previous work on Thespian, we presented an approach for applying directorial control while not affecting the consistency of characters' motivations. This work evaluates the effectiveness of our directorial control approach. Given the priority of generating only well-motivated characters' behaviors, we empirically evaluate how often the author's desired effects are achieved. We also discuss how the directorial control procedure can save the author effort in configuring the characters.

Categories and Subject Descriptors

J.4 [**Computer Applications**]: Social and Behavioral Sciences

General Terms

Experimentation

Keywords

Modeling narrative, Virtual agent applications and empirical studies, Agents in games and virtual environments, Directorial control

1. INTRODUCTION

Narrative is a central part of the human experience. With the rapid development of computer technology, a new form of media – interactive narrative – has received increasing attention [8, 14, 19, 5, 18, 3, 7, 27, 1, 10]. Interactive narrative allows the user to participate actively in a dynamically unfolding story, by playing a character or by exerting directorial control over events in the story.

By allowing the user to interact, interactive narrative provides a richer and potentially more engaging experience. Moreover, because different choices of the user lead to different paths through the story, the experience can be tailored for different users.

On the other hand, the support of user interactivity brings tremendous difficulty to the design process of interactive narratives. Unlike traditional narratives, where a single story line is presented, user interaction can lead to many alternative paths through the story. Within each path, the author typically wants the characters to act "within character", i.e. consistent with the characters' motivations. Further, the author often wants to achieve certain consistencies across the different story paths. For example, the author may want to create related story structures or dramatic moments for all the users. In reality, it is not even feasible to manually test each possible story path, not to mention design them.

1.1 Interactive Narrative Frameworks

To support rich user interactivity and prevent the author from spending extensive programming effort on hand-tailoring the narrative experience, various automated authoring frameworks for interactive narratives have been proposed. Most of the frameworks either adapt a plot-centric approach or a character-centric approach for modeling and simulating interactive narratives.

Plot-centric approaches for interactive narrative emphasize the design of the events in the story. In Façade [10], the story is organized around hand-authored dramatic beats. Based on a desired global plot arc, the drama manager chooses the next beat that is suitable to the context and whose dramatic value best matches the arc. In Mimesis [18], the authoring framework constructs story plans, which are the ideal linear narrative that the user should be told. When the user's action deviates from the story plan, the system either replans or prevents the user's action from being effective. The I-storytelling [3] system plans over a hierarchical tasks network (HTN) to realize interactive narratives. In IDA [7], stories are planned over SOAR-based agents. A drama manager is used to bring the story back on track if its development deviates from the ideal story path laid out by the author. In Lamstein's [6] and Nelson's [11] interactive narrative systems, the story is organized around events designed by the author, with pre- and post-conditions. A drama manager is used to project into the future for possible

Cite as: Evaluating Directorial Control in a Character-Centric Interactive Narrative Framework, Mei Si, Stacy C. Marsella and David V. Pynadath, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. XXX-XXX.

developments of the story, evaluate the quality of possible story paths based on an author-specified evaluation function, and reconfigure the story world to achieve best quality in the story. In [16], partial-order planning is used for generating the story and the author's directorial goals are incorporated as constraints on the generated results.

In contrast, a contemporary view on character and action, as espoused by Lajos Egri [4], suggests that plot unfolds based on the characters, that characters can essentially "plot their own story". Consistent with this view, character-centric approaches for interactive narrative emphasize the design of individually plausible characters. For example, FearNot! [14] is a planning based interactive narrative authoring and simulating system. It has explicit representations of characters' personalities and motivations, which affect the individual character's plan construction process. In MRE [26] and SASO [28], there is an extensive dialogue management subsystem in each character that incorporates explicit rules for dialogues. The agents have plans governing the coherence of their behaviors which take their goals and emotions into account.

Most of the contemporary interactive narrative frameworks provide systematic support of either the design of character or plot structure, but rarely both (IDA [7] and Mimesis [18, 29, 17] are notable exceptions.) Thus, it is up to the human author to ensure by design that the other component is satisfactory, which is a significant undertaking and impossible in many cases.

1.2 Motivation for This Work

To achieve the author's desired effects in the story when facing a variety of user interactions, automated directorial control that continuously adjusts the story based on the user's behaviors is often applied. Most existing work on directorial control is built within plot-centric frameworks for interactive narratives. For example, the plot-centric authoring frameworks described in Section 1.1 have this functionality.

However, as discussed in Section 1.1, these approaches emphasize how events should happen during the interaction, and do not have a systematic way to ensure that the characters have consistent and human-like motivations. The coherence of narrative, which requires the events in the story to be meaningfully connected in both temporal and causal ways [13], is crucial for ensuring that people can understand their experience [2, 12]. A key aspect of creating coherent narratives is that the characters' behaviors must be interpretable to the user. Inconsistencies in the characters' motivations may confuse the user. Further, without being able to understand his/her experience, the user may not experience the story as designed by the author.

Another limitation in most of existing works on directorial control is that the interactive narrative frameworks do not model the user explicitly (IDA [7] is a notable exception.) Therefore, directorial controls are often applied based on rules predefined by the author for a "standard user," and cannot be adaptive to individuals who may react to the events differently.

Moreover, many approaches for directorial control are reactive [6, 11, 18, 10] rather than proactive. Thus, the system reacts to the user's actions, but does not proactively predict the user's action and take interventions ahead of time. Arguably, a proactive approach can be more effective in achieving directorial control. However, to adapt a proactive approach, a model of the user is necessary.

Thespian is a multi-agent framework for interactive narrative that takes a character-centric approach [20, 19]. Decision-theoretic goal-based agents are used for controlling each character in the story, with the character's motivations encoded as the agent's goals. To make the characters socially-aware and human-like when interacting with the user, Thespian models social normative behaviors [21], emotions in social interactions [24] and "Theory of Mind" [20]. In addition, Thespian contains a model of the user [22], which allows the system to predict the user's behaviors and experience, and also simulate the user as a way to test the interactive narrative.

In our previous work, we presented an approach for applying directorial control without affecting the consistency of characters' motivations [23]. Thespian's directorial control is proactive and tightly tied to the model of the user. Its director agent projects into the future for detecting potential violations of the author's authoring intention (directorial goals), which is expressed as partial order or temporal constraints on key events in the story. An event can be either an action from a character or a belief of a character including the user. Once a potential violation is detected, the director agent explores alternative methods for tweaking the characters' behaviors and to achieve the directorial goals. In [23], we demonstrated examples of applying directorial control, but did not include empirical evaluations.

This work empirically evaluates how often the author's directorial goals are achieved given the priority of maintaining consistent characters' motivations. The evaluation was conducted using simulated users instead of human subjects. This enables us to systematically test the effectiveness of the directorial control approach when facing different user interaction styles. Finally, in this work we also discuss how automated directorial control can save the author's effort in configuring the characters.

2. EXAMPLE DOMAIN

In this paper, the "Little Red Riding Hood" story is used to demonstrate our approach for directorial control. The user plays the role of the wolf. The story starts as Little Red Riding Hood (Red) and the wolf meet each other on the outskirt of a wood while Red is on her way to Granny's house. The wolf has a mind to eat Red, but dares not because there are some woodcutters close by. The wolf, however, will eat Red at other locations where nobody is around. Moreover, if the wolf hears about Granny from Red, it will even go eat her. Meanwhile, the hunter is searching the wood for the wolf. Once the wolf is killed, people who were eaten by it can escape.

3. DIRECTORIAL CONTROL IN THESPIAN

This section provides an overview of the Thespian framework, and in particular, how directorial control is realized.

3.1 Overview of Thespian

Thespian is a multi-agent framework for authoring and simulating interactive narratives. Thespian is built upon PsychSim [9, 15], a multi-agent system for social simulation based on Partially Observable Markov Decision Problems (POMDPs) [25].

3.1.1 Socially Aware Characters

In Thespian, each character in the story is controlled by a decision-theoretic goal-based agent, with the character's motivations encoded as the agent's goals [20, 19]. Each agent has multiple and potentially competing goals, e.g. keeping safe vs. keeping others safe, that can have different relative importance or preferences. For example, the wolf character can have goals on keeping safe and preventing itself from starving, with the former goal ten times more important than the latter. If the importance of the wolf's goals is the other way around, i.e. it is much more important for the wolf to not feel hungry than to keep himself alive, the wolf will try to eat people regardless of the situation.

Thespian agents have recursive beliefs about self and others, e.g. the wolf's belief about Red's belief about the wolf's goals, which forms a "Theory of Mind". The agents choose their actions during the interaction using a bounded lookahead policy. They project limited steps into the future, considering not only their own actions, but also other characters' responses using their mental models of other characters, and their responses in return. The agents choose the action that receives the highest expected reward to proceed.

3.1.2 Model the User

The user is also modeled using a Thespian agent based on the character whom the user takes the role of. The basic assumption is that by playing the character, the user will adapt to the character's goals to certain degree. However, in modeling the user, not only the goals of the user's character need to be considered, but also the goals associated with game play, e.g. talking more (see [22] for more details).

For example, to model a user who plays the wolf character, we started by configuring an agent that models the wolf in the story. We then added two additional goals to the agent: to talk more and to explore the environment (by physically moving around) more. The relative importance of these two goals to the agent's other goals decides what type of user is being modeled, such as a very talkative user, a very shy user, or a user who sticks strictly to the character's goals.

The modeling of the user allows other agents to form mental models about the user in the same way as about other characters and the director agent to reason about the user's beliefs and experience during the interaction.

3.1.3 "Fitting" Characters' Motivations

To help the author set up the motivations of the characters, Thespian provides an automated fitting procedure that can constrain the agents' space of goals based on criteria generalized from examples of ideal story paths. The author provides ideal story paths by laying out sequences of the characters' including the user's interactions. Thespian's fitting procedure automatically extracts constraints on characters' motivations from story path examples and determines whether consistent goal preferences of agents explain the behavior. Usually the result of fitting is a space of possible goal settings, each of which can motivate the characters to act as the author specified in the ideal story paths. Using this procedure, the author can tune virtual characters by writing linear paths, and Thespian can generalize from these author-specified path examples to a larger space of possible paths under the constraints learned from the examples, i.e. the characters will use the motivations "learned" from the paths to interact with the user when the user deviates from the paths.

Table 1: Syntax for Specifying Directorial Goals

orders =	[*event*1,*event*2] *event*2 should happen after *event*1
earlierThan =	[*event*,*step*] *event* should happen before *step* steps of interaction
laterThan =	[*event*,*step*] *event* should happen after *step* steps of interaction
earlierThan2 =	[*event*1,*event*2,*step*] *event*2 should happen within *step* steps after *event*1 happened
laterThan2 =	[*event*1,*event*2,*step*] *event*2 should happen at least after *step* steps after *event*1 happened
NoObjIfLater =	[*event*,*step*] if *event* hasn't happen after *step* steps of interaction, the constraint for it to happen if exists, does not apply any more

3.2 Directorial Control

Thespian utilizes a specialized agent – a director agent – to realize directorial control. Unlike other agents, the director agent is not mapped to an on-screen character. The director agent also has accurate beliefs about all other agents including their goals, actions and beliefs about each other. In contrast, for modeling narratives it is often necessary for characters to have incorrect beliefs about each other.

When the director agent is functioning, it takes over other agents' decision-making process, decides the best movements for the story and causes other agents to perform the corresponding actions.

This section presents the syntax which the author can use for specifying directorial goals, and briefly describes the directorial control process, in particular, how the characters' motivations are kept consistent from the user's perspective during the process.

3.2.1 Directorial Goals

Directorial goals are used by the author to indicate how they want the story to progress, such as when an action should happen, or a character should change its belief about another character. Thespian supports directorial goals expressed as a combination of temporal and partial order constraints on the characters' including the user's actions and beliefs. Currently, six different types of goals are supported, as shown in Table 1. The directorial goals of a story can contain one or more goals of each type (see Table 4 for an example.)

The events in the syntax can be either an action, e.g. "wolf-eat-Granny" or a character's belief, e.g. "Red: wolf's hunger = 0 (Red believes that the value of the wolf's state feature hunger is 0)." "anybody" can be used in defining actions in directorial goals. It indicates that the corresponding field can be filled with any character, e.g. "anybody-kill-wolf".

3.2.2 Director Agent

The director agent works by projecting into the future and checking whether the future development of the story will be consistent with the author's directorial goals. By default, directorial control is applied every time after the user performs an action. This replaces the characters' decision-making process. If the director agent can foresee a violation, it will try to tweak the virtual characters' beliefs and behaviors to prevent the violation from happening. This process happens in the director agent's simulation of the future. It allows the director agent to perform adjustments to the virtual characters' configurations ahead of time to prevent the violation from happening.

The basic challenge in this process is how to ensure that the virtual characters exhibit consistent and interpretable motivations throughout the story while modifying their beliefs and behaviors. Further, the characters' motivations should be consistent with the author's portrayals of the characters in the story paths used for configuring (fitting) the agents.

The director agent uses fitting based algorithms to make the users feel that they are interacting with consistent characters. The basic assumption is that the user will not have a precise mental model about the characters because the user's observations in the story will not support such precision. Typically a range of configurations of a character can be used to explain the character's behaviors (recall that the result from fitting a character's motivations to a set of story paths is usually a range of goal weights.) Therefore, modifying a character's goals and beliefs does not necessarily lead to broken characters' motivations being experienced. The boundary of the range, i.e. how precise the user's mental model about the character is, is decided by their prior interactions and the user's observation of the character. In general, the more the two characters have interacted before, the more precise their mental models about each other are, and therefore the smaller the ranges are. In Thespian, a character's pre-existing beliefs and personality are modeled as the agent's initial beliefs and goals. As the characters, including the user, interact with each other, the agents' state update and belief revision processes refine their mental models. From the perspective of applying directorial control, the user will not experience inconsistency in the characters' motivations if the modifications to the characters fall within the user's mental models of the characters. For example, in this domain, the director agent can freely arrange the locations of characters whom the user cannot see. Recall that Thespian's fitting procedure tests whether consistent characters' motivations can be inferred from one or more story paths. A slightly modified fitting procedure is used in directorial control to filter out suggested changes to characters' behaviors that are not consistent with the characters' prior behaviors. Similarly, a fitting based procedure is applied for filtering out unrealistic suggestions for modifying the characters' beliefs. A suggestion is not realistic when the procedure can not find an explanation for the belief change (see [23] for details.)

4. EVALUATION ON THE EFFECTIVENESS OF DIRECTORIAL CONTROL IN THESPIAN

This section provides an evaluation on the effectiveness of Thespian's directorial control. A set of tests were conducted

Table 2: Directorial Goals

orders =	[[wolf-eat-Granny, anybody-kill-wolf], [red-give-cake-granny, wolf-eat-red]] [red-give-cake-granny, wolf-eat-granny]]
earlierThan =	[[wolf-enter-house, 90], [anybody-talkabout-granny-wolf, 30]]

Table 3: Conditions for Evaluation

	Talkative	Non-Talkative
Hunter Close	Condition I	Condition II
Hunter Far Away	Condition III	Condition IV

for investigating how often Thespian can successfully create the author's desired story when facing a variety of users and initial settings of the story, given the priority of consistent characters' motivations. To systematically generate the user's behaviors, the user was simulated using a Thespian agent.

4.1 Procedure

This evaluation compares the success rates of achieving directorial goals with and without the director agent. When the director agent is used, the directorial goals listed in Table 2 were set as its goals.

The user was simulated using a Thespian agent. This "user agent" plays the wolf in the story. Note that the directorial goals and the user's personal goals are different. The user, in most cases, does not know what kind of experience the author seeks and certainly will not purposely act to realize the author's design.

In this study, two common types of users were simulated: talkative users and non-talkative users. The agent simulating a talkative user regards talking more as a very important goal, only the goals of safety and not being hungry are more important than that. It will actively initiate conversations with other characters and try to maintain the conversation as far as its more important goals are not going to be hurt, e.g. the hunter is not close by. A non-talkative user also considers safety and not feeling hungry as its two most important goals. However, the importance of having more conversation is lower than some of its other goals, such as moving around and exploring the environment. As a result, it responds to others, but will not initiate a conversation.

The initial condition of the story includes the user and other characters' status and beliefs, such as where they are, and what they think their relationships with others are. These settings affect how likely certain events will happen in the story. In this study, the hunter's initial location was varied. The location was set to be either close to where the wolf, Red and the woodcutter were (these three characters were placed next to each other at the beginning of the story) or far away from them. In the latter case, the user has more chance to carry a long conversation with Red or the woodcutter at the beginning of the story.

In total, this study is conducted with four conditions, as listed in Table 3. For each condition, both interactions with and without the director agent are simulated. The director agent knows the initial condition of the story, but has no

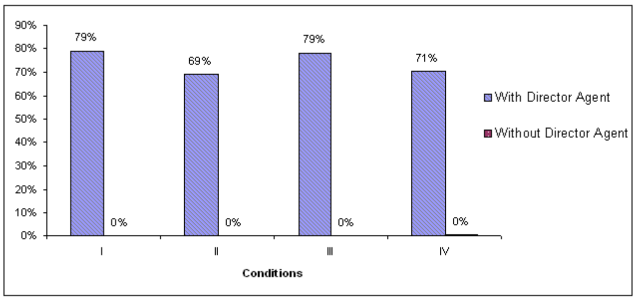

Figure 1: Success Rates of Directorial Control

information regarding what type of user is being simulated.

Each simulated interaction contains 25 rounds, i.e. both the user and the virtual characters act 25 times in the story without counting additional fictional actions or changes of beliefs resulting from directorial control.

Because the total number of interactions grows exponentially with the number of available choices the user has at each step, it is impossible to simulate all the interactions. In this study, the whole space of interactions was sampled 10 times. For each sample, 200 to 450 interactions were randomly simulated. The interactions were sampled as following. At the first n rounds of the interaction, the user agent randomly picked a small number of actions to simulate when it had multiple action choices that were all consistent with its motivations; and after the initial n rounds, the user agent only randomly chose one action to proceed (of course, this action has to be consistent with its motivation.) The number of actions to pick at each step and the number n were set based on experience to ensure that the total number of interactions falls within the range of [200,450].

Algorithm 1 shows the process of simulating the user. This process allows variations in the user agent's behaviors while keeping the agent's motivations consistent using a fitting based approach. Algorithm 1 is based on the algorithm used in [22] for simulating the user, with modifications for randomly sampling the user's behavior.

4.2 Effectiveness of Directorial Control

Figure 1 shows the percentages of simulated paths that satisfy the directorial goals in each condition by averaging the results from the 10 sets of simulations. The results from the simulations without the director agents serve as a base line. In this case, when the director agent was not used, what happened during the interaction was never consistent with the directorial goals. With the director agent, the success rates are around 70% to 80%, which is a large improvement over the base line.

When the director agent is not used, the user's experience is fully determined by the initial setting of the story and by how the user interacts. The users' experiences can be very diverse because there is a large space of parameters to be set by the author, i.e. the initial state and beliefs of each of the characters. When setting up the story, the author may not fully realize the long term impact of each setting. For example, it is hard to estimate how the hunter's initial location affects whether the wolf eats Red before Red gives the cake to Granny. Further, the user can interact with different styles, which also affects how the story will unfold.

In Figure 1, because the directorial goals are never reached,

Algorithm 1 GENERATE-ALL-PATHS(*user*, *maxstep*, *existPath*, *n*, *m*)

```
user : the name of the user character
maxstep : steps left to simulate
existPath : actions that have already happened
n: : after the initial n rounds, the user agent will only
  randomly chooses one action to proceed
m: : within the initial n rounds, the user agent will
  randomly chooses m action to proceed

allOptions ← []
curStep ← 25 - maxstep
for each action a in user.getOptions() do
  pathnew ← existPath.append(a)
  res ← Fit(user,pathnew)
  # if this is a possible path
  if res then
    allOptions = allOptions.append(a)

for each action a in allOptions do
  selected ← false
  if curStep ≤ n then
    if length(allOptions) ≤ m then
      selected ← true
    else
      # randomly select m options
      selected ← random(length(allOptions), m)
  else
    selected ← random(length(allOptions), 1)
  if ! selected then
    continue()
  # simulate the user does the action
  simulate(a)
  # simulate other characters' responses
  while user does not have the next turn do
    other_character's_action ← getResponse()
    pathnew ← pathnew.append(other_character's_action)
  maxstep_new ← maxstep - 1
  if maxstep_new > 0 then
    Generate-All-Paths(user,maxstep_new,pathnew,n,
      m )

Fit(user,pathnew): fits user's goals to pathnew
```

one cannot observe how the initial setting of the story and the user's interaction style affect the achievements of directorial goals. An additional set of experiments was performed to show the effects of these factors. The results are shown in Figure 2. In this additional study, the wolf's initial social distance to other characters was set to be slightly closer. This makes other characters more likely to tell the wolf Granny's location. Instead of sampling the space of possible interactions 10 times, only 1 sample for each condition is randomly drawn. Similarly, the numbers of interactions simulated for each condition is within the range of [200, 450].

Figure 2 shows that without the director agent, a talkative user is more likely to have an experience consistent with the directorial goals, especially when the hunter is initially placed far away from the user. However, if a different set of directorial goals was provided, different user interaction styles or initial configurations of the story may be favored. It is time-consuming to test these parameters. It may even be

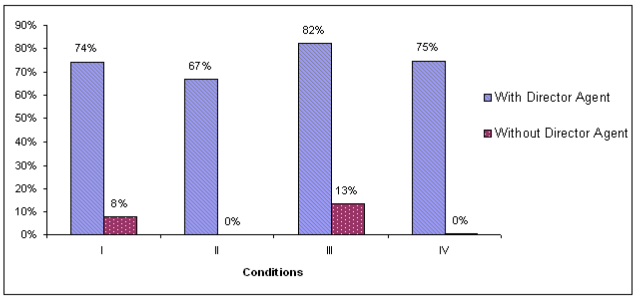

Figure 2: Success Rates When the Wolf Has Closer Social Distance with Others

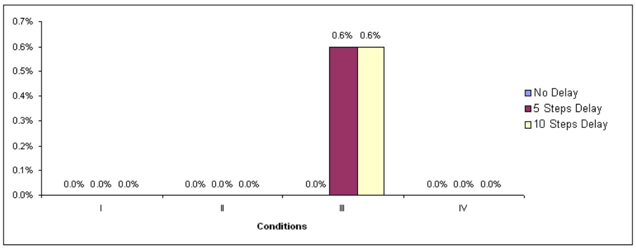

Figure 3: Delay in Achieving Directorial Goals without Director Agent

impossible to find an initial configuration of the story that ensures the directorial goals will be achieved for all kinds of users.

In general the director agent can tune the user's experience toward the directorial goals set by the author. This function, while keeping the user's control of the character (agency), makes the users' experiences less diverse and also less sensitive to the initial setting of the story and the user's interaction style. The author can thus avoid the time-consuming process of tweaking the initial setting of the story, and have their authoring effort greatly saved. As shown in Figure 1 and Figure 2, the director agent reached similar and high success rates in all conditions.

4.3 Delay in Achieving Directorial Goals

Directorial goals are sometimes achieved with a delay. The results showed in Figure 1 are the statistics of achieving directorial goals without any delay in time, for example, if a directorial goal specifies that "wolf-eat-Granny" should happen by the 20th step of the interaction, and if this action has not happened by that time, directorial control is considered failed for that story path. However, because of the director agent's attempts to make "wolf-eat-Granny" happen, even though the event may not happen on time, it may take place sometime after.

Figure 3 and Figure 4 show the statistics of achieving directorial goals within 5 steps and 10 steps of delay respectively. What can be observed is that the success rates of directorial control increase as longer delay is allowed. However, this only happens when the director agent is used. Without the director agent, the success rate of only one condition increased, and the increase is trivial. This serves as evidence for the director agent's effect on tuning the user's experience toward the directorial goals.

Multiple reasons can account for the delay. The most possible one is that the user, who is simulated by a Thespian agent, did not act exactly as the director agent expected. Thus, though the director agent has set up an environment for the author's desired effect to happen, it has to wait for the user to do the "right" actions.

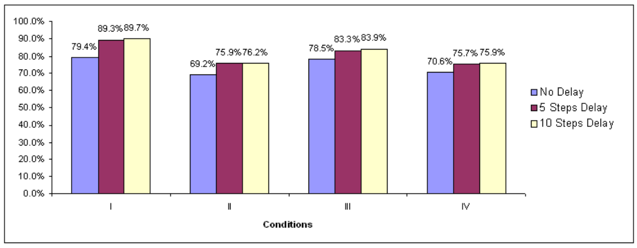

Figure 4: Delay in Achieving Directorial Goals with Director Agent

Table 4: Directorial Goals Variation I

orders =	[[wolf-eat-Granny, anybody-kill-wolf], [red-give-cake-granny, wolf-eat-red]] [red-give-cake-granny, wolf-eat-granny]]
earlierThan =	[[wolf-enter-house, 90], [anybody-talkabout-granny-wolf, 30]],
laterThan =	[[wolf-eat-Red, 90]]

4.4 Varying Directorial Goals

In previous evaluations, the same set of directorial goals is used. To test the generality of directorial control and study how the design of directorial goals can affect the effect of directorial control, two additional evaluations are performed with different variations of directorial goals.

4.4.1 Variation I

This evaluation tests the effectiveness of directorial control when the director agent is given the goals listed in Table 4. Compared to the goals in Table 2, this set of directorial goals has an additional temporal constraint: the wolf should not eat Red until the 90th step of the interaction.

Similar to the evaluations performed in Section 4.2, we sampled the user's interactions 10 times with and without the director agent. This evaluation was performed at a smaller scale. Each time, around 20 paths were sampled. In this evaluation, the user's interaction style and the initial setting of the hunter were not varied. A talkative user was simulated, and the hunter was placed close to the user's initial location.

The results of this evaluation show that without the director agent, only 17% of the interactions are consistent with the directorial goals on average. With the director agent, 81% of the interactions satisfy the directorial goals.

4.4.2 Variation II

The procedure for this evaluation is similar to that used in Section 4.4.1. The directorial goals listed in Table 5 are applied. This set of goals defines a slightly different style of story, in which the events happen in a more even pace.

In this evaluation, the directorial control fails. Only a few simulated paths satisfy the directorial goals. On a closer

Table 5: Directorial Goals Variation II

orders =	[[wolf-eat-Granny, anybody-kill-wolf]]
earlierThan =	[[wolf-enter-house, 120], [anybody-talkabout-granny-wolf, 60]],
laterThan =	[[wolf-eat-Red, 60]]
laterThan2 =	[(wolf-eat-red, wolf-enter-house, 10)]

examination, we find most of the paths failed to satisfy the following temporal constraint: "laterThan2 = [(wolf-eat-red, wolf-enter-house, 10)]". "wolf-enter-house" is required to happen before the 120th step. On the other hand, there is no directorial goal to ensure "wolf-eat-red" happens before that.

In general, the author may not fully realize the dependencies among different directorial goals when defining an interactive experience. In our future work, as discussed in Section 6, we plan to build automated procedures for helping the authors identify potential conflicts and inefficiencies among the directorial goals.

5. DISCUSSION

The results of these evaluations demonstrate the effectiveness of directorial control. Without the director agent, the generated story paths are diverse. They depend on the characters' and the user's motivations, and also on the initial settings of the story that are independent from the characters' motivations. When the director agent is present, the behaviors of the characters are more consistent with what has been described in the directorial goals.

It can also be observed that the interactions gradually converge to the author-specified directorial goals. When counting the number of paths that achieve directorial goals, if a time delay is allowed, many more paths achieve the goals when the director agent is used. In comparison, without the director agent, only a few more paths achieve the goals.

However, directorial control is not guaranteed to always be successful. When directorial control fails, it is usually because of two reasons. First, the directorial goals cannot be achieved under certain conditions. This is discussed in more detail in Section 6. Second, it is also possible that the director agent does not have enough time to respond after it detects a potential violation of directorial goals. The director agent can only foresee and plan limited steps ahead. Increasing the step limits will probably help achieving directorial goals. However, there is a trade-off because the further the director agent reasons about the future, the longer it will take to make a decision.

6. FUTURE WORK

Our future work will provide additional authoring tools to facilitate the authors in designing directorial goals. The author can define how they want the story to unfold using multiple partial order constraints and temporal constraints. A natural question to ask is how to set the appropriate constraints. In general, the constraints defined by the author may facilitate the achievements of each other, and may also conflict with each other, i.e. the achievement of one directorial goal prevent the achievement of another goal.

The conflicts among directorial goals can be either intrinsic to the set of goals or subject to the status of the interaction. An example of the former case is simple transitivity violations. For example, the author may specify that event A should happen before event B in one goal, event B should happen before event C in another goal, and event A should happen after event C in a third goal. The other type of conflict is contingent on the status of the story. Section 4.4.2 shows an example. Here is another example: the wolf needs to be killed within 20 steps after it eats Red, and also eat Granny within the first 50 steps of the interaction. If there is no constraint on when the wolf can eat Red, the wolf may eat Red at an early stage of the story, get killed, and never be able to eat Granny. Thus, the achievement of the first goal becomes an obstacle to the second. Whether this type of conflict happens depends on how the story unfolds, which is partially decided by the user. Currently, the author is allowed to deal with potential conflicts like this by specifying the priority among the goals by ordering them in descending order of importance. In the example provided in Section 4.4.2, priority was given to the earlierThan goals. Therefore, in most of the cases the wolf was able to find Granny's house and enter the house by the 120th step, but this action often happened without the wolf eating Red first.

We propose in our future work using automated approaches to identify potential conflicts in directorial goals, and providing the author suggestions on how to resolve the conflicts.

7. CONCLUSIONS

This work demonstrates the effectiveness of exerting directorial control in Thespian – a character-centric authoring framework for modeling and simulating interactive narratives. In Thespian, each character in the story is modeled as a decision-theoretic goal-based agent. Thespian ensures that the characters exhibit consistent motivations during the interaction, including when directorial control is applied. In addition, Thespian's directorial control is proactive and tightly tied to the model of the user. In comparison, most existing work on directorial control does not include a model of the user, and does not use a systematic approach to ensure the consistency of the characters' motivations.

To evaluate the effectiveness of Thespian's directorial control, we conducted a series of experiments. The results show that the directorial control procedure can effectively tune the development of the story toward a set of directorial goals defined by the author when facing different user interaction styles and initial settings of the story.

8. REFERENCES

[1] N. Braun. Storytelling in collaborative augmented reality environments. In *WSCG*, pages 33–40, 2003.

[2] J. Bruner. *Acts of Meaning.* Cambridge, Mass : Harvard University Press., 1990.

[3] M. Cavazza, F. Charles, and S. J. Mead. Agents' interaction in virtual storytelling. In *Proceedings of the International WorkShop on Intelligent Virtual Agents*, pages 156–170, 2001.

[4] L. Egri. *The Art of Dramatic Writing: Its Basis in the Creative Interpretation of Human Motives.* Simon & Schuster, 2004.

[5] M. T. Kelso, P. Weyhrauch, and J. Bates. Dramatic presence. *Presence: Teleoperators and Virtual Environments*, 2:1–15, 1993.

[6] A. Lamstein and M. Mateas. A search-based drama manager. In *AAAI Workshop Series: Challenges in Game Artificial Intelligence.*, 2004.

[7] B. Magerko and J. E. Laird. Mediating the tension between plot and interaction. In *AAAI Workshop Series: Challenges in Game Artificial Intelligence.*, pages 108–112, 2004.

[8] S. C. Marsella, W. L. Johnson, and C. Labore. Interactive pedagogical drama for health interventions. In *AIED*, pages 341–348, 2003.

[9] S. C. Marsella, D. V. Pynadath, and S. J. Read. PsychSim: Agent-based modeling of social interactions and influence. In *Proceedings of the International Conference on Cognitive Modeling*, pages 243–248, 2004.

[10] M. Mateas and A. Stern. Integrating plot, character and natural language processing in the interactive drama Façade. In *the International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, 2003.

[11] M. J. Nelson, D. L. Roberts, C. L. Isbell, and M. Mateas. Reinforcement learning for declarative optimization-based drama management. In *AAMAS*, 2006.

[12] E. Ochs and L. Capps. *Living narrative. Creating lives in everyday storytelling.* Cambridge, MA, Harvard University Press, 2001.

[13] S. Onega and J. A. G. Landa. *Narratology: An Introduction.* Longman, London and New York, 1996.

[14] A. Paiva, J. Dias, D. Sobral, R. Aylett, P. Sobreperez, S. Woods, and C. Zoll. Caring for agents and agents that care: Building empathic relations with synthetic agents. In *AAMAS*, pages 194–201, 2004.

[15] D. V. Pynadath and S. C. Marsella. Psychsim: Modeling theory of mind with decision-theoretic agents. In *IJCAI*, pages 1181–1186, 2005.

[16] M. O. Riedl. Incorporating authorial intent into generative narrative systems. In *AAAI Spring Symposium on Intelligent Narrative Technologies II*, 2009.

[17] M. O. Riedl and S. Andrew. Believable agents and intelligent scenario direction for social and cultural leadership training. In *the 15th Conference on Behavior Representation in Modeling and Simulation*, Baltimore, Maryland, 2006.

[18] M. O. Riedl, C. J. Saretto, and R. M. Young. Managing interaction between users and agents in a multi-agent storytelling environment. In *AAMAS*, pages 741–748, 2003.

[19] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: An architecture for interactive pedagogical drama. In *AIED*, pages 595–602, 2005.

[20] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: Using multi-agent fitting to craft interactive drama. In *AAMAS*, pages 21–28, 2005.

[21] M. Si, S. C. Marsella, and D. V. Pynadath. Thespian: Modeling socially normative behavior in a decision-theoretic framework. In *IVA*, pages 369–382, 2006.

[22] M. Si, S. C. Marsella, and D. V. Pynadath. Proactive authoring for interactive drama: An author's assistant. In *IVA*, pages 225–237, 2007.

[23] M. Si, S. C. Marsella, and D. V. Pynadath. Directorial control in a decision-theoretic framework for interactive narrative. In *Proceedings of International Conference on Interactive Digital Storytelling*, pages 221–233, Guimarães, Portugal, 2009.

[24] M. Si, S. C. Marsella, and D. V. Pynadath. Modeling appraisal in theory of mind reasoning. *JAAMAS.*, 2009.

[25] R. D. Smallwood and E. J. Sondik. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21:1071–1088, 1973.

[26] W. Swartout, R. Hill, J. Gratch, W. Johnson, C. Kyriakakis, C. LaBore, R. Lindheim, S. C. Marsella, D. Miraglia, B. Moore, J. Morie, J. Rickel, M. Thiúbaux, L. Tuch, R. Whitney, and J. Douglas. Toward the holodeck: Integrating graphics, sound, character and story. In *Agents*, pages 409–416, 2001.

[27] N. Szilas. IDtension: a narrative engine for interactive drama. In *the 1st International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, pages 14–25, March 2003.

[28] D. R. Traum, W. Swartout, S. C. Marsella, and J. Gratch. Fight, flight, or negotiate: Believable strategies for conversing under crisis. In *IVA*, 2005.

[29] R. M. Young, M. O. Riedl, M. Branly, A. H. Jhala, R. J. Martin, and C. J. Saretto. An architecture for integrating plan-based behavior generation with interactive game environments. In *Journal of Game Development.*, 2004.