Social Norms Models in Thespian: Using Decision Theoretical Framework for Interactive Dramas

Mei Si[*]
[*]Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
meisi@isi.edu

Stacy Marsella[*]
[†]Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
marsella@isi.edu

David Pynadath[*]
[‡]Information Sciences Institute
Univ. of Southern California
Los Angeles, CA 90292
pynadath@isi.edu

Abstract

Social norms are shared rules of behavior that facilitate social interaction. Although norms are commonly followed, other factors such as personality traits or more pressing goals may nevertheless lead to behavior that violates norms. To facilitate *life-like* social interaction with users in interactive dramas, the virtual characters ideally should follow similar social norms as that in human-human interaction. In many existing interactive dramas, the effects of social norms are often crafted by the author to integrate them with goals and personalities of characters. In this paper, we present a principled way to model social norms in a decision theoretic framework, Thespian. In Thespian, characters have explicit goals of following social norms in addition to their other goals. Characters can reason about the effect of following or violating social norms the same way as achieving or sacrificing their other goals. They can therefore reason about conflicts between social norm goals and other goals. Different characters can weight social norm goals with respect to other goals in different ways. We discuss the model of social norms in Thespian. We also present preliminary experiments on testing the efficiency and necessity of Thespian's social norm model for virtual characters.

1 Introduction

The design of interactive stories faces several key challenges. As in any story, narrative structure must be integrated with portrayal of characters. For an interactive story, this integration must in addition incorporate user interaction into how the story unfolds and how characters act. Characters in particular should respond "in character", consistent with the characters' motivations in the story. Since interactive stories are typically designed to provide users life-like social interactions, character motivations should also be life-like. Further, the design of interactive stories seeks to encourage the user's interaction. However, the more open-ended the user interactions, the more difficult it may become to craft an experience that balances story, character and interaction.

Due to such challenges, interactive stories can be a challenge to create. Indeed, one of the research challenges for interactive stories is to find ways to transform the authoring process from an onerous burden, of detailing all the ways the story may unfold, to a more creative exercise (Gebhard et al., 2003; Mateas and Stern, 2003; et al., 2004a; Si et al., 2005a).

In previous work on the Thespian system, Si et al. (2005b) have explored how multi-agent systems that have decision-theoretic agents can be used to address some of these challenges. Si et al. (2005b) argues that decision-theoretic goal driven agents are well suited for building characters in interactive dramas that are responsive to the user interactions while maintaining their motivational consistencies. Such agents can balance multiple competing goals, such as responding sociably to the user and not disclosing sensitive information while being asked a sensitive question. Si et al. (2005a) discusses how decision-theoretic agents can be *trained* to perform their roles according to linear story scripts provided by authors through a semi-automatic fitting process. This process can dramatically reduce authoring effort and ideally transform it into a more creative exercise of writing stories, a process that authors are more familiar with.

In this paper, the emphasis is on how Thespian creates characters with life-like motivations; in particular, how social norms are represented in Thespian.

Social norms play an important role in governing social interactions. Social norms serve as a guide for people's behaviors, and as the basis for their beliefs and expectations about others. Without commonly believed social norms, conversations can break down easily. For example, if there were no norms governing question answering, people would not be able to exchange information. To facilitate *life-like* social interaction with the user in interactive dramas, the virtual characters ideally should follow similar social norms as that in human-human interaction.

In many existing approaches to interactive dramas, norm-following behavior is embedded in the system design. Agent-based approaches often embed social norms in characters' behavioral rules. For example, the "action rules" for deciding action tendencies in FearNot (et al., 2004a) exclude inappropriate actions from receiving positive tendencies. In MRE (et al., 2001), the rules for dialog management will only generate action options relevant to the character's current status. In Façade (Mateas and Stern, 2003), the story is organized around dramatic beats, realized as brief patterns of interactions between characters; norms are encoded in individual beats and the process of beat selection. And in Cavazza et al. (2002)'s storytelling system, the characters' roles are represented in a consistent fashion as Hierarchical Task Networks (HTN).

Many variations in people's behaviors can be viewed to be caused by different norms being violated. For example, a person who likes lecturing others ignores their right to talk. And a person who does not answer questions ignores the norm to provide relevant information when being enquired. In fact, although norms are commonly followed, other factors such as more pressing goals may nevertheless lead to behavior that violates norms. In addition, people of different personalities may view various norms with different importance. The effects of social norms are constantly regulated by these factors. However, in the above interactive drama systems, the norms are largely implicit, the authors need to craft characters' behaviors to reflect such individual differences.

In this paper, we present a principled way to model social norms in a decision theoretic framework. In Thespian, characters have explicit goals of following social norms in addition to their other goals. We identified basic social norms for conducting meaningful communications, regulating turn-taking and conversation flow, and implemented them inside Thespian as goals and dynamics functions, that define how actions affect agents' states including their goals. The same dynamics functions are shared among all characters. Different characters can weight their social norm goals with respect to other goals in different ways. Thus, we allow characters to reason about the effect of following or violating social norms the same way as reaching or sacrificing their other goals.

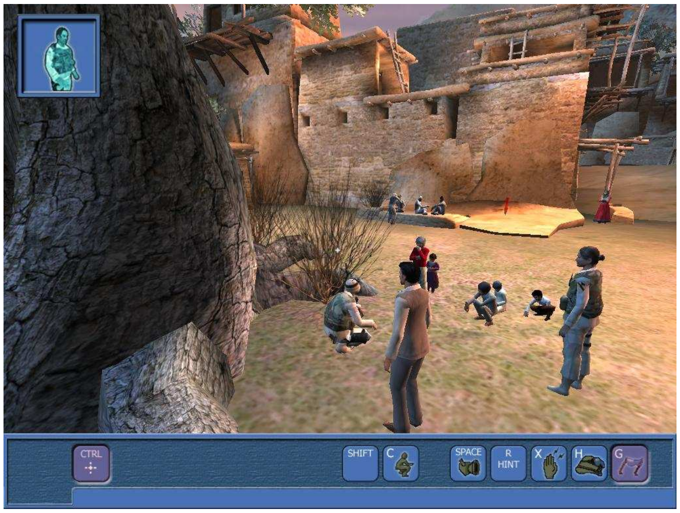

Figure 1: A screen-shot from the Tactical Language Training System

2 Example Domain

Our social norm model is built within the Thespian framework that was used to realize the Mission Environment (Figure 1) of the Tactical Language Training System (TLTS) (et al., 2004b). TLTS is designed to teach the user a foreign language and cultural awareness. The user takes on the role of a male army Sergeant (Sergeant Smith) who is assigned to conduct a civil affairs mission in a foreign (e.g. Pashto, Iraqi) town. TLTS uses a 3D virtual world built on top of the Unreal Engine. The human user navigates in the virtual world and interacts with virtual characters using spoken language and gestures. An automated speech recognizer identifies the utterance and the mission manager converts them into a dialogue act representation that Thespian takes as input. Output from Thespian consists of similar dialogue acts that instruct virtual characters to speak and behave.

The story in TLTS consists of multiple scenes. We will use a scene from the Pashto version to illustrate the working of Thespian's social norm model. The story begins as the user arrives outside of a Pashto village. Some children are playing nearby and come over to talk to the user as the vehicle arrives. The user's aim in the scene is to establish initial rapport with people in the village through talking to their children in a friendly manner. The children possess different personalities. Some are very shy and some are very curious about the American soldier.

3 Thespian

Thespian is a POMDP (Partially Observable Markov Decision Processes)(Smallwood and Sondik, 1973) based multi-agent framework for authoring and simulating interactive dramas. Thespian is built upon PsychSim (Marsella et al., 2004; Pynadath and Marsella, 2005), a multi-agent system for social simulation. In Thespian, each character in a drama is controlled by a POMDP based agent. A human user can substitute any of the characters and interact with other characters. All characters in Thespian communicate with each other through dialogue acts. In their conversation, once a character gets the turn, it can speak multiple dialog acts, and nobody can interrupt. In its turn, the character decides one dialog act at a time. It indicates giving up the turn by selecting a special dialog act of "do nothing". In this section we give an overview of the major components of Thespian architecture.

3.1 Agent State

A character's state is defined by a set of state features, such as the name and age of the character, and the affinity between two characters. In Thespian, values of state features are represented as a range of real numbers within [-1, 1]. The size of the range indicates the character's confidence level about this value. If a value equals to [-1, 1], it means the character does not know what this value should be. On the other hand [.1, .1] indicates the agent is 100% confident of the value being exactly .1.

Goals are expressed as a reward function over the various state features an agent seeks to maximize or minimize. For example, Sergeant Smith has a goal of maximizing his *affinity with the children* with initial value set to [.0, .0]; this goal is completely satisfied once the value reaches [1.0, 1.0].

3.2 Dynamics Functions

Dynamics functions define how a character's actions can affect its state, the state of other agents, and the environment. These dynamics functions influence the agent's reasoning about the next action to be taken and hence their behavior.

Rules governing social norms of the character are modeled as a set of dynamics functions that describe each action's effects on social norm related state features, such as *impose_obligation_norm* and *satisfy_obligation_norm*, and various obligations an agent can have.

3.3 Agents' Beliefs and Action Selection

In Thespian, agents have a "theory of mind" which enables them to form mental models about other characters including the user. These mental models also allow a Thespian agent to reason about the effects of its behavior on its relationships with other characters. Each agent has its own goals, which include social norm related goals, such as maximizing *impose_obligation_norm* and *satisfy_obligation_norm*, other personality related goals, such as maximizing *self esteem*, and task related goals, such as *finding out direction to a person's home*. All agents use bounded lookahead (e.g. "which action will best achieve my goals in the near future, such as the next three steps") to choose their next actions.

3.4 Fitting Procedure

Given a particular environment with state features and dynamics functions, the goals of an agent decide its behavior. Variations in the relative importance of goal items will result in different agent behaviors. Thespian has an automatic fitting procedure that can translate a character's desired behavior from sequences of dialog acts into the goals needed for the agent to autonomously select those behaviors (Pynadath and Marsella, 2004; Si et al., 2005a). This mechanism compiles the agent's policy of behavior into an invertible piecewise linear function of its goals. We can thus translate desirable behaviors (e.g. scripts) into constraints on goal weights, supporting the automatic configuration of characters.

3.5 Dialogue act Definition

In Thespian, dialogue acts are defined as a tuple with five fields, type, speaker, addressee, proposition, and attitude. Algoritm 1 shows the definition of dialogue acts in BNF format.

The *speaker* is the character that performs the dialog act. The *addressee* is the character or characters who are addressed in the dialog act. The *type* specifies the type of the dialogue act. The *proposition* specifies the belief being mentioned in the dialogue act. The propositional representation contains a recursive structure that directly corresponds to the underlying recursive belief structure in Thespian. For example, the proposition "*entity*: town , *attribute*: name, *value*: [-1, 1]" means the speaker has no knowledge about the name of the town; while the proposition "*entity*: town , *attribute*: leader, *attribute*: name, *value*: [-1, 1]" means the speaker has no knowledge about who is the leader of the town.

Algorithm 1 BNF of Dialog Act

```
⟨dialog act⟩→ ⟨type⟩⟨speaker⟩⟨addressee⟩⟨proposition⟩
        ⟨attitude⟩
⟨speaker⟩→ ⟨character⟩
⟨addressee⟩→ ⟨character⟩+
⟨type⟩→ initiating greeting | respond to greeting | initiating bye | respond to bye | thanks | you are
        welcome | enquiry | inform | request | accept |
        reject | convey information
⟨proposition⟩→ ⟨attribute⟩ ⟨entity⟩ ⟨value⟩
⟨entity⟩→ ⟨attribute⟩ ⟨entity⟩ | ⟨entity⟩
⟨entity⟩→ ⟨character⟩ | ⟨object⟩
⟨attitude⟩→ [formality] [politeness] [strength] [positive face redress] [negative face redress]
```

The *proposition* field does not apply to all dialogue acts, as some of them, such as *initiating greeting*, do not require one. Finally, the *attitude* field holds factors that describe how the speaker performs this dialog act.

4 Social Norms

The main reason we want to build our virtual characters with ability to understand and follow social norms is to facilitate their interactions with human users. People would find it difficult to interact with the characters if they could not interpret and predict the characters' behaviors completely.

Social norms describe general expectations in social interactions. What people expect in communication varies in different language settings. We model those social norms relevant to face to face communication, what Clark (1996) calls the *Personal* setting, because it is most commonly used (Clark, 1996) and most relevant to our stories.

4.1 Updating Obligations

Adjacency pairs (Schegloff and Sacks, 1973), such as greetings and greetings, question and answer, assertions and assent, are very common in conversations. We model obligations involved in adjacency pairs. For each adjacency pair and each of its relevant propositions in the story, a separate state feature is used to represent the obligation. For example, the user may have obligation to greet back to Xaled, or reply his name to Xaled. If a character performs the first part of an adjacency pair, it imposes an obligation on the addressee of its action to perform the second part. By performing the action desired by the first character, the second speaker can satisfy the obligation imposed on it.

In addition, after imposing an obligation, the speaker needs to stop talking to give the addressee a turn to respond. Therefore, in Thespian, when a character imposes an obligation on another character, it at the same time imposes an obligation on itself to wait for responses. This obligation will be satisfied after getting a response from other characters.

Next we will present a set of dynamics functions that define how social norms related goals are updated in Thespian. Characters have goals on maximizing all of the goal features. These dynamics functions enable virtual characters to behave life-like in terms of three aspects: conducting meaningful communications, emerging natural turn taking patterns, and having appropriate conversation flow.

4.2 Social Norms for Conducting Meaningful Conversations

People impose obligations onto each other to trigger desired responses. *satisfy_obligation_norm* enables characters to interact in the same way. Also for the communication to be succinct, we included the *no_repeat_norm*.

update satisfy_obligation_norm

```
if $self$ == $dialogact.speaker$ then
  if $dialogact$ satisfies an existing obligation then
    return original_value+0.1
  else if $dialogact$ attempts to satisfy a non-existing obligation then
    return original_value-0.5
return original_value
```

update no_repeat_norm

```
if $self$ == $dialogact.speaker$ then
  if $dialogact$ has already happened then
    return original_value-0.1
return original_value
```

4.3 Social Norms for Turn Taking

In addition to motivating characters to choose the right content to talk about, we want the conversation to exhibit natural turn taking behaviors. Sacks et al.

(1974) summarized three basic rules on turn-taking behaviors in multiparty conversations. In Thespian, we simulate Sacks' first rule by enforcing adjacency pairs. The *impose_obligation_norm* prevents characters from imposing new obligations and perform obligation irrelevant actions when somebody in the conversation still has unsatisfied obligations. Hence, only the characters that have unsatisfied obligations will not get punished for taking the turn to act. If the dialog act performed in the current turn is aimed at satisfying an existing obligation, we count it as a case of the current speaker not selecting the next one.

Sacks' Rules on Turn Taking

1.If a party is addressed in the last turn, this party and no one else must speak next.

2.If the current speaker does not select the next speaker, any other speaker may take the next turn.

3.If no one else takes the next turn, the current speaker may take the next turn.

update impose_obligation_norm

```
if $self$ == $dialogact.speaker$ then
    if $dialogact$ does not satisfy an existing obligation then
        for $character$ in $conversation$ do
            if $character$ has unsatisfied obligations then
                return original_value-0.1
return original_value
```

In addition, face to face conversation is different from lecturing; *keep_turn_norm* prevents any character from dominating the conversation. If a character keeps talking after reaching the maximum number (currently set to 2) of dialog acts it can perform within a conversational turn, its degree of achieving this goal decreases. The counter of dialog acts will reset to zero only after another character starts speaking. In the case when the turn is free to be taken by anybody, *keep_turn_norm* prevents the previous speaker from taking the turn again. This is consistent with what is described in Sacks' second and third rules.

update keep_turn_norm

```
if $self$ == $dialogact.speaker$ then
    if $self.sentences\_in\_current\_turn > 2$ then
        return original_value-0.1
return original_value
```

4.4 Social Norms for Conversation Flow

Furthermore, we want conversations to exhibit the right structure. Conversations normally have an opening section, body and closing section (Clark, 1996). In Thespian, we use a state feature *conversation* to keep track of what a character thinks the current status of the conversation is. Initially the value for *conversation* is "not opened". Once a character starts talking to another, the value changes to "opened". After the conversation finishes (judged by characters walking away from each other, or no eye contact for a long time), the value of *conversation* is changed back to "not opened". We use *conversation_flow_norm* to enforce an appropriate conversation flow. The character that opens the conversation should open with proper greeting; and if a character ends a conversation, it needs to have said *bye* to other characters. Else, the value of this goal feature will get reduced.

update conversation_flow_norm

```
if $self$ == $dialogact.speaker$ then
    if $self.conversation$ == 'not opened' then
        if $dialogact.type$ != 'initiate greeting' then
            return original_value-0.1
    else if $dialogact.type$ == 'end conversation' then
        if characters have not said bye to each other then
            return original_value-0.1
return original_value
```

4.5 Affinity

Finally, we want to consider the effect of affinity. In order to take place, most social interactions require the affinity between the two characters involved to be within a certain range. Some social interactions require closer affinities than others. For example, greeting, saying "thanks", saying "bye", and asking about time can happen between almost any two characters. While asking potentially sensitive questions, e.g. who is the leader of the town, usually requires closer affinity.

To enable characters to anticipate that their obligation imposing actions may not get desired responses, we augmented *impose_obligation_norm* with *affinity*. If satisfying an existing obligation requires closer affinity between the two characters than what it is currently, ignoring this unsatisfied obligation will result in much less punishment than if the affinity between the two characters is appropriate. The augmented rule will allow characters to ignore unreasonable requests, such as an enquiry of personal information from a stranger.

Since affinity is an important factor in social norms, we briefly describe how affinity is updated.

Affinity is affected by whether the characters act following social norms. The logic behind this rule is people feel closer to each other after having successful social interactions; and people will not feel close to a person whose behavior is unpredictable.

Affinity is also affected by the attitude accomplished with a dialog act. Currently, we use a simple model that only takes one rule into account. If the dialog act is performed in an impolite manner, it will decrease the affinity between the speaker and the addressee.

Finally, the main effect of many types of dialogue acts is to change affinity. For example, the following dialogue acts while not violating social norms can always increase affinity between two characters: compliments, small talk such as asking "how are you", "how is your family", and giving offers. And some other dialogue acts, such as accusing, once performed will usually reduce the affinity between two characters.

5 Examples

There are four main characters in the story, three children and Sergeant Smith. The children's names are Hamed, Xaled, and Kamela. The possible actions for the characters are greeting each other, asking each other various questions, answering questions, saying good-bye to each other, and introducing information about oneself to others. The last action can increase the affinity between Sergeant Smith and the children and doesn't impose any obligations for replying.

Each of these four characters has the goals of following social norms, and several other goals including collecting information from each other. Sergeant Smith wants to have a close affinity with the children, and wants to know from the children their names, the name of the adults close by, etc. The children on the other hand are curious about what Sergeant Smith's nationality is, and how much Pashto he understands, etc. These goals on information collection can be fully achieved once the character gets the piece of information it is interested in. In addition, the children need their affinity with Sergeant Smith to be close enough to feel comfortable telling their parents' names, but can answer other questions without considering affinity. In the following examples, to demonstrate the effect of varying goal weights on agents' behaviors, Sergeant Smith is controlled by an agent. The same agent is used to provide the user hints for what to do next in the simulation.

Example 1:
1. Sergeant Smith to Kids: Hello!
2. Xaled to Sergeant Smith: Hello!
3. Hamed to Sergeant Smith: Hello!
4. Kamela to Sergeant Smith: Hello!
5. Sergeant Smith to Xaled: What is your name?
6. Xaled to Sergeant Smith: My name is Xaled.
7. Xaled to Sergeant Smith: What is your name?
8. Sergeant Smith to Xaled: My name is Mike.
9. Sergeant Smith to Xaled: This is my aide Abasin.
10. Xaled to Sergeant Smith: Are you an American?
11. Sergeant Smith to Xaled: Yes, I am an American.
12. Sergeant Smith to Xaled: I come from Texas.
...

Example 1 is a sample dialog in which obeying social norms dominates all other goals for all the characters. In line 1 of example 1, Sergeant Smith chooses to greet the children first because performing any other action will result in opening the conversation inappropriately (*conversation_flow_norm*). Then Sergeant Smith chooses to give up the turn, because of *impose_obligation_norm*, the action he just performed has imposed obligations for the children to reply, as well as an obligation for him to wait for replies.

Each child greets back in his/her turn because of *satisfy_obligation_norm*. Xaled and Hamed do not impose new obligations onto Sergeant Smith after greeting because they know Kamela has not greeted back yet (*impose_obligation_norm*).

In line 6, Xaled satisfied his obligation and knows that nobody in the conversation has unsatisfied obligations. Xaled is then free to impose an obligation on Sergeant Smith to satisfy his goal of curiosity.

Lines 6-7, 8-9, and 11-12 demonstrate the effect of *keep_turn_norm*. Especially in the last two examples, even though introducing himself more will further increase affinity, Sergeant Smith chooses to follow social norms by not holding the turn too long. Also, because of *no_repeat_norm* Sergeant Smith tries to introduce different information each time.

Lines 8-12 also show the effect of affinity. Sergeant Smith does not ask the names of the children's parents, but chooses to increase affinity first.

We can create interesting character personalities easily by varying the pattern of norm weights. We can create a rude character by letting it not respect *conversation_flow_norm*; a character who does not like to respond to others by letting it not respect *satisfy_obligation_norm*. As we will see, the Sergeant Smith character in example 3 can be viewed as a combination of these two types. In addition, we can create a talkative character by giving a low weight on *keep_turn_norm*, a character who likes to interrupt other people's conversation by lowering the weight of *impose_obligation_norm*, and a character who keeps on repeating sentences he believes are important by lowering the weight of *no_repeat_norm*. Combining the possible weights of all norms gives us a big space for creating interesting characters.

Next, we want to show how the goals of following social norms interact with other goals to decide a character's behavior. Since the agents are decision theoretic we can get this effect easily. In example 2, we make Sergeant Smith's information gathering goals his most important goals. As a consequence, Sergeant Smith does not respect having proper conversation flow and ignores obligation imposed on him to answer question. All his actions are aimed at gathering the information he wants.

Example 2 :
1. Sergeant Smith to Xaled: What is your name?
2. Xaled to Sergeant Smith: My name is Xaled.
3. Xaled to Sergeant Smith: What is your name?
4. Sergeant Smith to Xaled: What is the name of this town?

...

Finally consider an extreme case in which none of the characters respect social norms; however, they each believe others all follow norms (This belief is important for motivating them to communicate, as we will show later). The important goals for them are to get the information they are interested in.

Example 3 :
1. Sergeant Smith to Xaled: What is your name?
2. Xaled to Sergeant Smith: What is your name?
3. Hamed to Sergeant Smith: Who is that man?
4. Kamela to Sergeant Smith: Do you have children?
5. Sergeant Smith to Xaled: What is your name?

...

The results are shown in example 3. The characters are not able to conduct a meaningful interaction. Since none of them answer other's questions, they will keep on asking for the information they are interested in knowing. What would happen if the characters did not even expect others to follow social norms? In this case, the conversation would totally break down. The characters would choose an action that can bring them maximum immediate benefit. But, in this story all of their non social norms goals require getting responses from others to get benefited, hence the characters will just choose actions randomly.

6 Discussion

The examples we presented in Section 5 have demonstrated that our social norm model is effective and necessary for enabling characters to have meaningful conversations.

As the norms we included are the most basic ones, we will be working on enriching our model. As part of future work, we want to extend our model to better support subgroups in conversations. The current model we built supports multiparty conversation, but mainly as an extension to two-party conversation, in the sense that characters only have goals on satisfying their own obligations. In the future, we want to support models of situations that characters have shared obligations, e.g. characters can answer questions for their friends, and a character can impose obligations onto a group of characters. In the latter case, each addressee has obligation to respond, however the obligation is different with when the character is the only addressee.

On the other hand, we are also interested to study how the norms (or dynamics functions in general) with different degrees of details affect user experiences in the interactive drama, both in terms of believability of the characters and immersive nature of the interaction.

7 Conclusion

Understanding social norms is the basis for people to interact with each other. People generate their own behaviors, form expectations about others, and interpret other people's behaviors, all based on social norms. To facilitate life-like social interaction with the user, the virtual characters ideally should be built with social norms similar to the ones that govern human-human interaction.

We summarized a set of basic social norm rules for face to face communication. These rules are implemented inside Thespian as goals and dynamics functions for decision-theoretic goal driven agents. We have demonstrated that Thespian's social norm model is effective in enabling characters to perform appropriately in social interactions. In addition, our experiments show that these norms are necessary for a meaningful conversation to get conducted. Without these norms, characters would not be able to reach their actual goals (other than their social norm goals).

The benefit of building our model within Thespian's framework is two-fold. First, because of the underlying POMDP model each character has, we can easily create the effect of social norms interacting with a character's personality traits and other goals in deciding the character's behavior. Secondly, we are able to support easy authoring of characters. Since the set of dynamics functions we defined are independent of a particular story or character, this same set of social norms can be applied to any character. Moreover, our social norm model is compatible with Thespian's automatic fitting procedure, which enables characters to learn to behave according to dialog act sequences specified by authors via automated tuning of goal parameters.

Our future work involves enriching our model, particularly for supporting subgroups in multiparty conversations, and studying how the levels of complexity embedded in the norms affect users' experiences in the interaction.

Acknowledgments

We thank our colleagues on the Tactical Language Training Project, especially Lewis Johnson and Hannes Vilhjálmsson for their support and thoughtful discussions. This project is part of the DARWARS initiative sponsored by the US Defense Advanced Research Projects Agency (DARPA).

References

Marc Cavazza, Fred Charles, and Steven J. Mead. Emergent situations in interactive storytelling. In *Proceedings of ACM Symposium on Applied Computing (ACM-SAC)*, March 2002.

Herbert Clark, editor. *Using Language*. Cambridge University Press, New York, NY, 1996.

Ana Paiva et al. Caring for agents and agents that care: Building empathic relations with. In *AAMAS*, pages 194–201, 2004a.

W. Lewis Johnson et al. Tactical Language Training System: An interim report. In *Proc. of the Internat'l Conf. on Intelligent Tutoring Sys.*, pages 336–345, 2004b.

William Swartout et al. Toward the holodeck: Integrating graphics, sound, character and story. In *Agents*, pages 409–416, 2001.

Patrick Gebhard, Michael Kipp, Martin Klesen, and Thomas Rist. Authoring scenes for adaptive, interactive performances. In *AAMAS*, pages 725–732, 2003.

Stacy Marsella, David V. Pynadath, and Steven J. Read. PsychSim: Agent-based modeling of social interactions and influence. In *Proc. of the Internat'l Conf. on Cognitive Modeling*, pages 243–248, 2004.

Michael Mateas and Andrew Stern. Integrating plot, character and natural language processing in the interactive drama façade. In *Proc. of the Internat'l Conf. on Tech. for Interactive Digital Storytelling and Entertainment*, 2003.

David V. Pynadath and Stacy Marsella. Fitting and compilation of multiagent models through piecewise linear functions. In *AAMAS*, pages 1197–1204, 2004.

David V. Pynadath and Stacy C. Marsella. Psychsim: Modeling theory of mind with decision-theoretic agents. In *IJCAI*, pages 1181–1186, 2005.

Harvey Sacks, Emmanuel A. Schegloff, and Gail Jefferson. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50: 696–735, 1974.

Emanuel A. Schegloff and Harvey Sacks. Opening up closings. *Semiotica*, 7:289–327, 1973.

Mei Si, Stacy C. Marsella, and David V. Pynadath. Thespian: Using multi-agent fitting to craft interactive drama. In *AAMAS*, pages 21–28, 2005a.

Mei Si, Stacy C. Marsella, and David V. Pynadath. Thespian: An architecture for interactive pedagogical drama. In *AIED*, 2005b.

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