Thespian: Modeling Socially Normative Behavior in a Decision-Theoretic Framework

Mei Si, Stacy C. Marsella, and David V. Pynadath

Information Sciences Institute
University of Southern California
Marina del Rey, CA 90292
meisi@isi.edu, marsella@isi.edu, pynadath@isi.edu

Abstract. To facilitate *lifelike* conversations with the human players in interactive dramas, virtual characters should follow similar conversational norms as those that govern human-human conversations. In this paper, we present a model of conversational norms in a decision-theoretic framework. This model is employed in the Thespian interactive drama system. In Thespian, characters have explicit goals of following norms, in addition to their other personal goals, and use a unified decision-theoretic framework to reason about conflicts among these goals. Different characters can weigh their goals in different ways and therefore have different behaviors. We discuss the model of conversational norms in Thespian. We also present preliminary experiments on modeling various kinds of characters using this model.

1 Introduction

Interactive dramas allow people to participate actively in a dynamically unfolding story, by playing a character or by exerting directorial control. There has been a growing research interest in computer-based, animated interactive dramas, both for entertainment (e.g., [1–3]) and for learning environments (e.g., [4, 5]), in part because their design faces a range of research challenges. Ideally, the user's interaction should be facilitated and they should have a sense that they can openly interact with the story. However consistency of story and character becomes harder to maintain in the face of open-ended user interaction. Addressing these challenges can lead to complex designs for interactive dramas, raising an additional challenge of how to facilitate authoring of the drama.

Our approach to these challenges is realized in an interactive drama system called Thespian. Characters in Thespian are realized as goal-driven, decision-theoretic agents that are responsive to the user's interaction while maintaining consistency with their roles in the story (see [6] for a discussion). The decision-theoretic framework allows them to balance multiple competing goals, such as responding sociably to the user but not disclosing sensitive information in a conversation. In prior work, we also demonstrated how the goal-driven agents can be *trained* to perform their roles, based on story scripts provided by authors that are then passed through a semi-automatic fitting process [7]. This process can reduce authoring effort compared to hand-authoring all possible interactions and ideally transform the authoring process into a more creative (and a more familiar) exercise of writing stories. In this paper, we focus on how Thespian agents model norms in conversations. Much as they do in human-human interaction, norm-following behaviors can facilitate and constrain user interactions in natural/lifelike ways that ideally do not seem restrictive.

In general, social norms are commonly believed rules in social interaction. These rules serve as a guide for human behavior, and as the basis for their beliefs and expectations about others. Without them, communication can break down easily. Though norms are commonly followed, the tendency to follow norms is regulated by other factors, such as more pressing, personal goals. There is a considerable body of work on social norms and norms in conversations in particular, including formalization of norms and obligations [8], how norms emerge, spread and get enforced in a society [9], levels of cooperation in social communications [10], discourse obligations in dialogues [11], maxims in cooperative conversations [12], etc.

Interactive dramas have taken differing approaches to incorporate norm-following in their designs. Norm-following/violating behavior is often not explicitly modeled. Rather, they are modeled conjointly with characters' other behaviors. In FearNot [5], affectively-driven characters are used. Characters' actions are either reactive to their current emotional states or result from the planning process using their internal goals, which are affected by their emotional states. In Façade [13], the story is organized around hand-authored dramatic beats, realized as their pre-conditions, post-conditions, and brief patterns of interactions between characters. Norms are encoded in interactions within beats and the beat selection process which is affected by the pre-conditions and post-conditions of the beats. In Cavazza's storytelling system [3], characters' behaviors are embedded in Hierarchical Task Network (HTN) plans crafted by authors. In SASO [14], there is an extensive dialogue management subsystem that incorporates explicit rules for normative behaviors, specifically conversational norms. The priorities of these rules are adjusted by agent authors to fit the characters' profiles. The modeling of a character's task and conversation are distinct but coupled.

We present a model of conversational norms crafted in Thespian's decision-theoretic framework. Thespian models several basic norms specific to face-to-face communication [15]. Conversational norms enable the characters to behave human-like in terms of three aspects: making relevant responses, following natural turn-taking patterns, and having appropriate conversational flow. Characters (the goal-driven agents) have explicit goals of following norms in addition to their other goals. Thus, we allow characters to reason about the effect of following or violating norms and achieving or sacrificing their other goals using a unified decision-theoretic framework. Moreover, the weights of goals can be automatically tuned using Thespian's fitting process.

In this paper, we discuss Thespian's conversational norms model in detail. We also illustrate its application to the Tactical Language Training System (TLTS) [16] for rapidly teaching students the rudiments of a foreign language and culture.

2 Example Domain

Fig. 1. A screen-shot from the Tactical Language Training System

Our conversational norms model is built within the Thespian framework that was used to realize the Mission Environment (Figure 1) of TLTS. 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 town. The TLTS uses a 3D virtual world built on top of the Unreal Tournament 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 what to say and how to behave.

We will use one of the scenes from the Pashto version to illustrate the working of Thespian's conversational norms 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

We developed Thespian as a multi-agent system for controlling virtual characters in an interactive drama. Thespian is built upon PsychSim [17, 18], a multi-agent system for social simulation based on Partially Observable Markov Decision Problems (POMDPs) [19].

Thespian's basic architecture uses POMDP based agents to control each character, with the character's personality and motivations encoded as agent goals. Objects (e.g., a town, a house) in the story can also be represented as special Thespian agents that only have state features, but not actions, goals, policy and beliefs about others, to enable characters to reason about the values of their state features in the same way as those of a character. All characters communicate with each other through dialogue acts. A human user can substitute for any of the characters and interact with others.

3.1 Thespian Agent

This section describes the components of a Thespian agent, including its state, beliefs, action dynamics functions, goals, and policies.

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. Values of state features are represented as a range of real numbers within [-1, 1] (alphabetic values are encoded as real values using a constant convention through out the scene.) For example, an agent's name, Mike, might be encoded as .1. There is usually uncertainty involved in agents' beliefs about other agents' and/or its own state features. The size of the range indicates the character's confidence level about this value. For example, if a character believes another character's name is [-1, 1], it means the character does not know the name. On the other hand [.1, .1] indicates the agent is 100% confident of the value being exactly .1.

Beliefs The agent's subjective view of the world includes its beliefs about itself and other agents and *their* subjective views of the world, a form of recursive agent modeling [20]. An agent's subjective view about itself or another agent can include every component of that agent, such as state, beliefs, policy, etc.

Dynamics Dynamics functions define how actions can affect agents' states. For example, greetings among agents set the state feature *conversation status* of all participants to [1.0, 1.0], indicating a conversation among them has started (for more complex examples, see the dynamics defined in the Conversational Norms section.) Dynamics functions in an agent's belief space define how this agent believes an action will affect agents' states. Therefore dynamics functions influence the agent's reasoning about what action to take and hence its behavior.

Goals We model a character's personality profile as its various goals and their relative importance (weight).

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]. An agent usually has multiple goals with different relative importance (weights). For example, Sergeant Smith may have another goal of knowing the children's names, and this goal may be twice as important to him as the previous goal. The weights of the goals decide what action the agent will choose given the same dynamics.

Policy In Thespian, all agents use a bounded lookahead policy. Each agent has a set of candidate actions to choose from when making decisions. When an agent selects its next action, it projects into the future to evaluate the effect of each option on the state and beliefs of other entities in the story. The agent considers not just the immediate effect, but also the expected responses of other characters and, in turn, the effects of those responses, and its reaction to those responses and so on. The agent evaluates the overall effect with respect to its goals and then chooses the action that has the highest expected value.

Consider the following simplified example from the story to illustrate the agent's lookahead process. Before the conversation starts, Sergeant Smith considers what he should say first. He can either greet the children or ask the children a question he cares about. Greeting does not affect the value of any of his goals; starting a conversation without greeting will hurt his goal of maintaining a normal conversational flow. In the next step, the responses he expects from the children are "greeting back" and answering the question respectively. Getting the answer from the children will satisfy his goal of obtaining that piece of information, and "greeting back" will not affect his state. What action Sergeant Smith will choose to do depends on the relative importance of maintaining a normal conversational flow versus obtaining that piece of information.

In the above example, only two steps of the interactions are considered by the agent. Theoretically, each agent can perform lookahead for large enough number of steps until there is no gain for itself and other agents. For performance reasons, we limit the lookahead to a finite horizon that we determine to be sufficiently realistic without incurring too much computational overhead (e.g., for the examples in this paper, the horizon is three times the total number of characters in the conversation).

3.2 Fitting procedure

To craft an interactive experience, the author can either configure the characters' goal weights by hand to ensure they behave appropriately or use the fitting procedure for help. To use the fitting procedure, the author needs to define the characters' roles in the story by creating alternative linear scripts (sequences of dialogue acts) of the desired paths of the story. Using the scripts as constraints on allowable agents' behaviors, the fitting process [21, 7] can tune agents' goal weights so that they behave according to the scripts.

Before running the fitting procedure, the author sets the initial conditions including the goal weights for all of the characters. By default these initial values will be used to set the initial beliefs characters have about each other. The goal weights do not necessarily need to be accurate, since the fitting process will automatically adjust them.

In fitting, Thespian proceeds iteratively for each story path, fitting the goals of one agent at a time and holding all other agents' goals as fixed. For each action in the story path, if the action is performed by the agent that is currently being fitted, the fitting process simulates the agent's lookahead process, and calculates constraints on goal weights to ensure that the desired action receives highest utility among all candidate actions. So in the earlier example, Sergeant Smith needs to have a higher weight on maintaining a normal conversational flow than obtaining the information to ensure that he chooses to greet first.

At the end of the fitting process, the constraints resulting from fitting each path can be merged into one common constraint set. Typically, there are multiple candidate goal weight values that are consistent with the preferred story paths. Thespian can pick one of these solutions according to its own heuristics, which is to choose the goal weights as close to the original ones as possible. It also gives the author the option of manually selecting one from the constrained set.

A character's goal weights after fitting are usually different from their initial values set by the author. This difference can lead to discrepancies between a character's actual personality and another character's mental model of it. The author can synchronize the models by repeating the fitting step with the agents' beliefs set to the actual personality. However, characters do not necessarily have to have exact knowledge about other characters or themselves to exhibit the desired behaviors. In fact, it can be dramatically interesting when characters do not have accurate models of each other.

4 Conversational Norms

Thespian's conversational norms model consists of goals that motivate characters to behave socially appropriately, state features that keep track of the status of conversation, affinity among characters and obligations each character has, and dynamics functions for updating these state features. Characters have goals to maximize all of their goal features.

4.1 Adjacency Pairs

Adjacency pairs[22], such as greet and greet back, enquiry and inform are very common in conversations. They are performed by two speakers and follow a fixed pattern. We use an obligation-based approach to model this social phenomenon. Obligations are represented by agents' state features. Figure 2 lists some of the adjacency pairs we model currently and the obligations related to them. The character that performs the first part of an adjacency pair creates an obligation for the addressee to perform the second part. By performing the action desired by the first speaker, the second speaker can satisfy the obligation. For example, if Sergeant Smith opens the conversation by greeting the children, the children have obligations to greet Sergeant Smith back, in which case the values of the corresponding state features are set to [1.0, 1.0]; and once the obligations are satisfied, the values will go back to its default level of [0.0, 0.0], indicating the children do not have such obligations. After creating an obligation for the addressee, the first speaker needs to stop talking to give the addressee a turn to respond. To motivate characters to do so, an obligation of waiting for responses is created by the first speaker for itself. This obligation will be satisfied after getting a response from other characters.

Speaker 1	**Speaker 2**	**Obligation**
Greet	Greet back	Greet back to *speaker 1*
Bye	Bye	Say "Bye" to *speaker 1*
Thanks	You are welcome	Say "You are welcome" to *speaker 1*
Offer X	Accept/Reject X	Either accept or reject X to *speaker 1*
Request X	Accept/Reject X	Either accept or reject X to *speaker 1*
Enquiry about X	Inform about X	Inform to *speaker 1* about X
Inform information	Acknowledgement	Acknowledgement to *speaker 1*

Fig. 2. Adjacency Pairs and Corresponding Obligations

To enforce adjacency pairs, we give each character a goal of maximizing its state feature *complete_adjacency_pair_norm*. If the agent's dialogue act satisfies one of its obligations, the value of this state feature will increase. If the dialogue act intends to satisfy an obligation[1], but the agent does not actually have such an obligation (for example, the agent says "you are welcome" when nobody has said "thanks" to it), the value of this state feature will decrease. The specific amounts of increase or decrease only have relative meaning when they are for the same state feature. For example, in Algorithm 1, we believe violating the norm is more serious than following it[2].

Algorithm 1 Dynamics for complete_adjacency_pair_norm

```
if self == dialogueact.speaker then
  if dialogueact intends to satisfy an obligation then
    if the agent has such obligation then
      return original_value+0.1
    else
      return original_value-0.5
return original_value
```

[1] Note that the communicative intent of a dialogue act is explicit in Thespian at this level. So we are always able to tell if the character is trying to create or satisfy an obligation.

[2] It is meaningless to compare the amounts of changes on different state features, since the fitting process will scale the agent's goal weights and therefore counter this difference.

4.2 Turn Taking

In addition to motivating characters to complete adjacency pairs, we want their conversation to exhibit natural turn-taking behaviors. Sacks et al. summarized three basic rules on turn-taking behaviors in multiparty conversations [23]:

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.

In Thespian, we use the goal state feature *initiate_adjacency_pair_norm* to keep track of how appropriate it is for a character to create obligations for others. The character's degree of achieving this goal will reduce if it creates new obligations for others when somebody in the conversation still has obligations. Hence, under this circumstance, only the characters that have obligations will not get punished for seizing the turn to act. If the dialogue 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 speaker.

Algorithm 2 Dynamics for initiate_adjacency_pair_norm

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

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

4.3 Conversational Flow

We want conversations to exhibit the right structure. Conversations normally have an opening section, body and closing section [22]. In Thespian, we use a

Algorithm 3 Dynamics for keep_turn_norm

```
if self == dialogueact.speaker then
  if self.sentences_in_current_turn >2 then
    return original_value-0.1
return original_value
```

state feature *conversation status* to keep track of what a character thinks the current status of the conversation is. Initially the value for *conversation status* 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 status* is changed back to "not opened". We use the goal of maximizing *conversational_flow_norm* to enforce an appropriate conversational 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. Otherwise, the value of this goal feature will get reduced.

Algorithm 4 Dynamics for conversational_flow_norm

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

4.4 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", and asking about time can happen between almost any two characters. While asking private or potentially sensitive questions, e.g. who is the leader of the town, closer affinity is required.

To enable characters to anticipate that their actions may not trigger desired responses, we augmented *initiate_adjacency_pair_norm* with *affinity*. If satisfying an obligation requires closer affinity between the two characters than what it is currently, ignoring this 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. And because characters have models of each other, the enquirer will know his/her enquiry is unreasonable and may be ignored.

Affinity is affected by many factors. First, it is affected by whether the characters act following norms. In Thespian, characters are closer to each other after having successful social interactions; and if a character constantly violates norms, its affinity with other characters will decrease. Affinity is also affected by the attitude associated with a dialogue act. Currently, we use a simple model that only takes one rule into account. If the dialogue 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, when not violating 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 accusations, once performed will usually reduce the affinity between two characters.

5 Example Dialogues

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, small talk, and introducing information about oneself to others. The last action can increase the affinity between Sergeant Smith and the children and does not create any obligations for replying.

Each of these four characters has the goals of following 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 the children's names, the names 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 corresponding piece of information. 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. However, normally a human learner would play Sergeant Smith, in which case the agent could be used to provide hints to the learner about what to do next. Even though in the actual authoring process, characters' goal weights are often fitted to their desired behaviors defined by the author, in the following examples, we will directly manipulate characters' goal weights to show the possible range of behaviors our model can create.

Example 1 is a sample dialogue in which obeying 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 (hurting his goal of *conversational_flow_norm*). Then Sergeant Smith chooses to give up the turn, because of his goal of maximizing *initiate_adjacency_pair_norm*. The action he just performed has cre-

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: How are you?
10. Xaled to Sergeant Smith: I am fine.
11. Xaled to Sergeant Smith: Are you an American?
12. Sergeant Smith to Xaled: Yes, I am an American.
13. Sergeant Smith to Xaled: I am learning Pashto.

...

ated 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 their *complete_adjacency_pair_norm* goals. Xaled and Hamed stop talking after greeting because they know Kamela has not greeted back yet; if they create obligations for others, their *initiate_adjacency_pair_norm* goals will be hurt. In line 7, Xaled has satisfied his obligation and knows that nobody in the conversation has obligations. Xaled is then free to ask Sergeant Smith questions to satisfy his goal of curiosity. Lines 6-7, 8-9, 10-11, and 12-13 demonstrate the effect of the *keep_turn_norm* goal. Especially in lines 12-13, even though introducing himself more will further increase affinity, Sergeant Smith chooses to follow norms by not holding the turn too long. Lines 8-13 also show the effect of affinity. Sergeant Smith does not ask the names of the children's parents directly, but chooses to talk about other topics to increase the affinity between them first.

We can create various character personalities by varying the pattern of norm weights. For example, we can create a character that seems either rude or in a hurry by letting it not respect *conversational_flow_norm*; a character who does not like to respond to others by letting it not respect *complete_adjacency_pair_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 *initiate_adjacency_pair_norm*. The above examples show varying the weights of different norms gives us a large space for creating different characters. The author does not need to worry about the correspondence between goal weights patterns and characters' personalities, because the fitting process can automatically configure characters' goal weights given scripts describing their behaviors.

Next, we want to show how the goals of following norms interact with other goals to decide a character's behavior. Because 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

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?
. . .

does not respect having proper conversational flow and ignores his obligation to answer question. All his actions are aimed at gathering the information.

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?
. . .

Example 3 shows an extreme case in which none of the characters respect norms. However, they believe others will follow norms. The important goals for them are to get the information they are interested in. The characters are not able to conduct a meaningful interaction. Since none of them answer questions, they keep on asking for the information they are interested in.

What would happen if the characters did not even expect others to follow 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 norms goals require getting responses from others to get benefit; hence the characters will just choose actions randomly.

6 Discussion and Future Work

The examples we presented in Section 5 have shown that our conversational norms model is capable of modeling various kinds of characters in social communication. This model has been applied to 11 characters in three TLTS scenes, which consists of 65 lines on an average.

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. We want to support modeling 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.

On the other hand, we are also interested to study how the norms (or action dynamics in general) modeled 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.

In addition, the evaluation of this work is currently primitive. As future work, we would like to develop more formal methodology for evaluating the system.

7 Conclusion

We discussed a model of basic conversational norms for face to face communication. These norms are implemented inside Thespian as goals and dynamics functions for decision-theoretic goal-driven agents. We have demonstrated that Thespian's conversational norms model is capable of modeling various kinds of characters in social interactions.

The benefit of building our model within Thespian's framework is three-fold. First, because of the underlying POMDP model each character has, we can easily create the effect of norms interacting with a character's other goals in deciding the character's behavior. Secondly, since the dynamics functions are independent of the characters (their goals, beliefs), this same model can be applied to any character. Finally, the approach is consistent with the automated authoring in Thespian, which enables characters to be tuned to behave according to dialogue act sequences specified by authors via automated tuning of goal parameters.

Our future work involves enriching our model and developing evaluation methodologies. For enriching the model, we are particularly interested in 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, especially Lewis Johnson, Hannes Vilhjálmsson and David Traum 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

1. Kelso, M.T., Weyhrauch, P., Bates, J.: Dramatic presence. PRESENCE: Teleoperators and Virtual Environments **2(1)** (1993) 1–15
2. Riedl, M., Saretto, C.J., Young, R.M.: Managing interaction between users and agents in a multi-agent storytelling environment. In: AAMAS. (2003) 741–748
3. Cavazza, M., Charles, F., Mead, S.J.: Emergent situations in interactive storytelling. In: Proc. of ACM Symposium on Applied Computing (ACM-SAC). (2002)
4. Marsella, S.C., Johnson, W.L., Labore, C.: Interactive pedagogical drama for health interventions. In: AIED. (2003)
5. Paiva, A., Dias, J., Sobral, D., Aylett, R.: Caring for agents and agents that care: Building empathic relations with. In: AAMAS. (2004) 194–201
6. Si, M., Marsella, S.C., Pynadath, D.V.: Thespian: An architecture for interactive pedagogical drama. In: AIED. (2005)

7. Si, M., Marsella, S.C., Pynadath, D.V.: Thespian: Using multi-agent fitting to craft interactive drama. In: AAMAS. (2005) 21–28
8. Boella, G., Torre, L.v.d.: Obligations as social constructs. In: Proc. of the Italian Conf. on Artificial Intelligence (AI*IA'03). (2003) 27–38
9. Castelfranchi, C.: Commitments: From individual intentions to groups and organizations. In: ICMAS. (1995) 41–48
10. Airenti, G., Bara, B.G., Colombetti, M.: Conversation and bahavior games in the pragmatics of dialogue. In: Cognitive Science. Volume 17(2). (1993) 197–256–48
11. Traum, D.R., Allen, J.F.: Discourse obligations in dialogue processing. In: ACL. (1994) 1–8
12. Grice, H.P.: Logic and conversation. In Cole, P., Morgan, J., eds.: Syntax and Semantics: Vol. 3: Speech Acts. Academic Press (1975) 41–58
13. Mateas, M., Stern, A.: 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)
14. Traum, D.R., Swartout, W., Marsella, S.C., Gratch, J.: Fight, flight, or negotiate: Believable strategies for conversing under crisis. In: IVA. (2005)
15. Clark, H., ed.: Using Language. Cambridge University Press, New York, NY (1996)
16. Johnson, W.L., Beal, C., Fowles-Winkler, A., Lauper, U., Marsella, S.C., Narayanan, S., Papachristou, D., Vilhjálmsson, H.H.: Tactical Language Training System: An interim report. In: Proc. of the Internat'l Conf. on Intelligent Tutoring Sys. (2004) 336–345
17. Marsella, S.C., Pynadath, D.V., Read, S.J.: PsychSim: Agent-based modeling of social interactions and influence. In: Proc. of the Internat'l Conf. on Cognitive Modeling. (2004) 243–248
18. Pynadath, D.V., Marsella, S.C.: Psychsim: Modeling theory of mind with decision-theoretic agents. In: IJCAI. (2005) 1181–1186
19. Smallwood, R.D., Sondik, E.J.: The optimal control of partially observable Markov processes over a finite horizon. Operations Research **21** (1973) 1071–1088
20. Gmytrasiewicz, P., Durfee, E.: A rigorous, operational formalization of recursive modeling. In: ICMAS. (1995) 125–132
21. Pynadath, D.V., Marsella, S.C.: Fitting and compilation of multiagent models through piecewise linear functions. In: AAMAS. (2004) 1197–1204
22. Schegloff, E.A., Sacks, H.: Opening up closings. Semiotica **7** (1973) 289–327
23. Sacks, H., Schegloff, E.A., Jefferson, G.: A simplest systematics for the organization of turn-taking for conversation. Language **50** (1974) 696–735