AAAI Technical Report WS-11-18

A Rule-Based Framework for Modular Development of In-Game Interactive Dialogue Simulation

Emmett Tomai, David Salinas and Rosendo Salazar
[1]Department of Computer Science, University of Texas - Pan American, 1201 West University Dr.
Edinburg, TX 78539 USA
(etomai@cs.panam.edu)

Abstract

In this paper, we discuss approaches to dialogue in interactive video games and interactive narrative research. We propose that situating interactive dialogue in the simplified expectations of video games is a profitable way to investigate computational dialogue simulation. Taking cues from existing physical simulations such as combat, we propose a hypothetical game environment and design goals for an embedded interactive dialogue system. We present a modular framework targeted at that environment, which is designed to enable incremental development and exploration of dialogue concepts. We describe this framework together with a work-in-progress system for simulating simple in-game negotiation dialogues.

Introduction

An increasing number of high-budget commercial video games are experimenting with alternative interactive dialogue mechanisms. These changes can be broadly categorized into novel user interfaces and increased impact of dialogue choices on the ongoing story. BioWare, a long-time industry leader in the Role-Playing Game (RPG) genre, has produced several iterations of interactive dialogue in their *Dragon Age* and *Mass Effect* series, and will be releasing another system in the upcoming *Star Wars: The Old Republic*. Quantic Dream's *Heavy Rain*, played as an interactive movie, taking the player through many different variations of scenes and outcomes. And Rockstar Games' recent *L.A. Noire* implemented a cinematic police interrogation game as a major part of the gameplay. This trend is notable for research in interactive narrative, which seeks to make story and dialogue just as interactive as physical action.

Dialogue Interactions

Video games are far and away the most developed form of computationally driven interactive experience. However, games have largely separated the interactive elements (physical or physical-metaphor simulations) from the narrative elements (blocks of expository text, cut scenes, in-game descriptive text, etc) (Costikyan 2007). There is certainly a technical barrier to simulating those narrative elements, which research in interactive narrative seeks to overcome. However, considering the successful models of interaction in video games, exactly how simulated narrative would fit in is not as clear as it may seem. The obvious ideal is a living world, where the player interacts as fully as they would with the real world. But the real world is not an ideal narrative experience. There is lots of mundane, irrelevant, dead time. Movies provide a vision of fixing that, but how do you interactively "live out" a montage? Video games have their own pace and rhythm, which is not like real life or movies, books or plays. At the heart of almost every video game are repetitive encounters, and notable game designer Raph Koster has claimed that iterative mastering of the game mechanics in successively more complex contexts is exactly what makes games fun (Koster 2004). Specifically considering dialogue interaction, should it be more like a movie, or more like a game?

The majority of work in interactive narrative research, and in mainstream video games, has approached dialogue from the narrative point of view, as a story-driving encounter. In this work, we approach it from the interactive point of view, as a game. There are several interesting distinctions between well understood (and well received) physical interaction systems (e.g. combat, platform jumping, puzzles with physical elements or metaphors) and typical video game dialogue systems.

Dialogue and Interaction in Video Games

Most simulated interactions in games have clear win and loss conditions. Opponents are meant to be defeated or avoided, puzzles are meant to be solved, and pits, platforms and other obstacles are meant to be traversed without dying. When a loss condition occurs, the most common result is that the player must retry the encounter(s) until he or she reaches the win condition. There are exceptions, of course, notably in creative and exploratory activities.

In contrast, dialogue interactions rarely have clear win and loss conditions. Many have no conditions at all, or serve only as exploratory hubs to click through exposition. In cases where the dialogue can alter relationships with NPCs, such as BioWare's RPGs, the user may have self-determined goals as to how those relationships will progress. In cases where the dialogue impacts the flow of the story (essentially plugging in to a higher-level branching structure), different paths may be considered more or less desirable and may again be the focus of user goals. Neither maps well to the typical model of winning and losing, which adds tension and challenge to games.

To support win/loss outcomes, most game interactions have clear feedback on the impact of actions towards success or failure. If you punch or shoot your opponent in games, their health meter declines, or they act like they've been hit, or they show physical deterioration. This is clear progress where the goal is to defeat, incapacitate or kill. Good puzzles tend to give some manner of logical feedback, even if it only in the mind of the player, otherwise they degrade into guessing games. When you eliminate a line in Tetris, you can see that you are further from the threatening ceiling. When you go through space-bending acrobatics to manipulate a switch in Valve's Portal games and a cube falls out, you assume that cube represents progress. Outcomes are predictable, allowing the player to improve decision-making over time.

This is difficult to implement in dialogue, which has considerably more hidden variables. BioWare's recent interfaces have switched from presenting choices as full sentences to shorter, simpler choices combined with fixed qualities (aggressiveness in Mass Effect 2) or thematic indicators (in Dragon Age 2). This strategy has been criticized when players "fill in the gaps" around that simplified choice and are surprised when it results in their character delivering a line that they did not expect. While game systems typically walk the player through progressive learning of the underlying mechanics, it would be difficult if not impossible for a writer to create dialogue interactions that follow a consistent underlying system of recognizable and predictable mechanics.

An Environment for Simulated Dialogue

We propose a hypothetical game environment for a system that simulates dialogue interactions. We believe that working within the established constraints of interactive games will allow us to create user experiences the can be effectively evaluated at both limited and expanded scale.

Encounter Design

Dialogue occurs in discrete encounters. Each encounter is between the player character (PC) and a single non-player character (NPC). Each turn in the dialogue consists of an utterance from one agent directed to the other. We adopt the most common approach of time-independent alternating turns. The following three goals constrain the interaction.

First, the dialogue must have tangible goals, and a clear, straightforward account of success and failure that the player can adopt. To meet this design goal, we have chosen to work with negotiation dialogues. These dialogues are situated in a hypothetical game using an industry standard RPG quest system. In such a system, the player character initiates more-or-less independent interactions with NPCs throughout the game world. Each dialogue interaction presents a contract of tasks and rewards, and has only two outcomes: the player chooses to either accept or reject the contract. In this system, the terms of each quest contract can be negotiated via the dialogue interaction, giving the player a clear and simple set of goals, for which success and failure can be readily evaluated. Within the quest framework, every dialogue begins with the NPC proposing a contract.

Second, the dialogue must be managed by a consistent simulation, and the player must be given per-choice feedback that indicates how the underlying state is being updated. The goal of this feedback is to allow the player to learn to recognize situations and actions that increase the chances of successfully meeting goals. In order to meet this design goal, there must be consistency in the contexts and actions of the dialogue. In any tractable design, this means repetition of utterances. To address this problem, we are exploring the use of descriptive rather than explicit dialogue. Using descriptive statements limits the space of linguistic forms that must be supported, but more importantly, we believe it will set player expectations so as to be more tolerant to repetition in the text. It is similar to the display of combat history in many RPGs, and players are used to this format to add detail to the primary graphical representation of the action. A simple example negotiation, reported in this *combat log* style, is shown in Figure 1.

```
Walrus Man wants you to kill 6 Snow Moose.

You request payment of 20 gold.

Walrus Man offers 10 gold.

You will take 15 gold.

Walrus Man agrees.
```

Figure 1. A simple descriptive dialogue.

Third, the dialogue choices will be generated and constrained by the simulation. The player will not have the freedom or burden of writing his or her own lines, and the NPC agents will have the same options that the player has (when in the same situation). Those choices must first be relevant to the current context, and second must give a range of interesting outcomes. One of the notable marks of narrative progression is that the characters just happen to make choices that result in an interesting story.

Given the use of descriptive text, the obvious way to present choices to the player is using the same descriptive utterances. But even with descriptive text, there is a great deal of nuance and ambiguity that is possible. Considering the second line of the example dialogue in Figure 1, the player character could as easily "demand payment" or "inquire about payment", with different intentions and interpretations. Further, the line as stated could be heard as an aggressive or humble move, as there is no indication in the surface text. Prior research has indicated that short, abstract choices can be ambiguous to players (Sali et al. 2010) with regard to the author's intentions. As in commercial games, this may be mitigated by decorating the choices with qualitative indicators of attitude or tone.

User Interface

We are remaining as interface agnostic as possible, beyond the commitment to descriptive text utterances. Independent of the interface particulars, the system must display a snapshot of the dialogue state that is updated each turn. This includes the history of descriptive utterances, as shown in Figure 1. It also includes a representation of the contract that is being negotiated. This contract must have two equal sides that list the obligations proposed for the NPC and the PC. It does not need to show the history of the contract, only the current state. Each obligation in the contract has a weight value, such that each side can be summed, and the entire contract be said to be balanced, or unbalanced in favor of one agent over the other. These weights and the overall balance must be displayed to the player. During the player character's turn, the interface must display the set of choices available to the player, as descriptive text, with any decorations. At various times the player should also have the option to accept the contract as it stands. The player should always have the option to reject the contract and walk away. Enforcing constraints or penalties on doing so is outside the dialogue system.

Negotiating

All changes to the contract must be *proposed*, and all proposals can be *accepted* or *countered* with a new proposal. Changes include adding and removing obligations and altering obligation parameters. Possible obligations are limited to common quest-type actions that the player can perform, such as hunting enemy agents, collecting items or visiting another NPC, and common rewards such as monetary payment and game items. Some of those obligations have a numeric parameter that can be modified in the negotiation, while others do not. One key element of negotiation that we are omitting here is uncertainly over the perceived value of an obligation. The "fair value" of the potential obligations is established a priori to the interaction by the author of the encounter, and that value is reflected in the weight of the obligation. This allows the system to generate meaningful choices, rather than having the player guess at numbers.

Multi-Layered Interpretation

One of the central challenges of dialogue is that the literal surface form of an utterance may not directly encode the actions that are implied by it. A request for payment, in the context of negotiating a contract, should be interpreted as a proposal – a modification of the contract subject to the approval of the other party. Because the proposal is self-serving, it should also be interpreted as an aggressive move, which shifts the tone towards competitive rather than cooperative bargaining. And this counter might cause the originator to consider the countering agent less of an ally, or take offense, or develop enmity or various other emotional reactions. Interpretation of an utterance must consider these different concerns against the context of the setting, the situation and the characters. In a real-world application, the number of such concerns is overwhelming, but in a game we can restrict which concerns matter according to how they impact the game state and user experience. In this work we focused on aspects of contract negotiation, and the interaction with concerns about character *reputation*.

Many RPG quest systems are tied to a system of reputation, an alternate currency that ostensibly represents an NPC group's positive, self-interested opinion of the player character. This simple abstraction of a complex interpersonal system provides a good starting point to expand the impact of negotiation dialogue beyond the bounds of the negotiation itself. Each player move in the dialogue can result in a gain of loss of reputation, reflecting the NPC agent's opinion of the player's tactics. This reputation must be displayed to the player, and updated with each turn to immediately reflect gains or losses.

Related Work

Early work by the Oz group at Carnegie Mellon extended the HAP architecture for believable agents to use natural language (Loyall & Bates 1997). This system generated dialogue that reflected agent goals, personality, emotional state and what the agents are aware of in their environment. The reactive architecture could combine action and dialogue acts to reach goals, and produce believable real-time effects such as self-interruption.

More recent work has taken a more plot-centric approach, under the broad challenge of integrating believable agent behavior with authorial goals. The interactive drama Façade (Mateas and Stern 2002) revolves almost entirely around dialogue between virtual actors, with the player intervening in ongoing interactions via natural language. The system uses reactive planning with joint authored goals to present believable characters in a coherent, directed plot with interactive freedom. With such unconstrained boundaries in user input, interaction failures are inevitable, and evaluations of the system have focused on the way players react and adapt (Knickmeyer & Mateas 2005; Mehta et al. 2007). One notable finding is that clarifications of the system's understanding only serve to undermine the player's ability to project coherence.

Cavazza and Charles have investigated the integration of dialogue into narrative structure, recognizing that dialogue must believable as communication between actors, but also evaluated rhetorically as part of the narrative presentation (Cavazza and Charles 2005). They have implemented dialogues between AI characters centering on social event planning, supporting both explicit communication and implicit speech acts. In their system, character is the unifying model that keeps actions and dialogue on the same narrative track.

The Virtual Humans project has created a very sophisticated model of negotiation dialogue for use in military training simulations (Traum et al. 2008). The goal in that work is specifically realistic interactions, as opposed to the wider range of experiential goals in interactive narrative as a whole.

The obvious alternative interactive dialogue system in the gaming industry is in Maxis' TheSims franchise. This system uses hierarchical menus of descriptive dialogue acts such as "Gossip" and "Tell a joke" to present user dialogue choices. The dialogue itself is rendered as a special gibberish language, and combat-style feedback is provided for each action and the current state. This interface is significantly limited to abstract conversations. The player is not tied to any dialogue structure, turn-taking or timing, and even basic response pairs are not supported. This suggests that it would be difficult convey either plot or character through these interactions.

A notable research system that explores dialogue as a game is *Comme il Faut (CiF)* (McCoy et al. 2010). In this system, the entire game is based on a simulation of social relationships, which is updated via social interaction games. The point of the game as a whole is to explore these social interactions and the characters that play them out, creating an emergent plot. We are pursuing a parallel interest in dialogue games using descriptive choices, but in the context of existing, task-oriented game dynamics. As we work towards integration of character relationships, the CiF simulation of social relationships will certainly be very relevant work.

Framework for Dialogue Simulation

The dialogue is broken into alternating turns between the two agents, where each turn consists of a dialogue move. The dialogue move is a set of frames, representing the dialogue actions performed by the character utterance for that turn. The dialogue move must have exactly one surface frame, representing the literal form of the utterance. It also has zero or more action frames that specify additional dialogue actions that are entailed by the surface frame in that turn.

type	surface
action	desire
agent	npc
aggressiveness	neutral
object	
type	surface
action	do-infinitive
agent	player
object	
type	obligation
class	action
action	kill
target	[Snow Moose]
magnitude	10

Figure 2. Surface frame for killing 10 Snow Moose.

type	action
action	propose
agent	npc
aggressiveness	neutral
target	
type	proposal
action	add
side	npc
partial	true
target	
type	obligation
class	action
action	kill
target	[Snow Moose]
magnitude	10

Figure 3. Proposal action frame for killing 10 Snow Moose.

Example frames for the first line in Figure 1 ("Walrus Man wants you to kill 6 Snow Moose ") are given in Figure 2 and Figure 3.

Inference rules axiomatize these frames with respect to a dialogue move and the simulation state before and after that move. Rules may infer that:

(1) Given a set of actions in a move and a prior conditional state, another set of actions are entailed in that move
(2) Given a surface in a move and a prior conditional state, a set of actions are entailed in that move
(3) Given a set of actions in a move and a prior conditional state, a set of state changes are entailed
(4) Given a prior conditional state, a set of actions is entailed as a relevant part of the next move
(5) Given a prior conditional state, a set of frames abstracting that state are entailed

We have chosen to represent the state of the dialogue following the current-snapshot approach found in typical game simulations. In such an environment, rules are expected to consider only the current state, and any aggregate trends must be an explicit part of that state. The complete representation of prior moves is also maintained, making reconstruction of prior states possible, but we want to incentivize rule and simulation construction to avoid that. We are pursuing this course as the most appropriate for embedding in a video game.

Reasoning Over Dialogue Choices

The system performs three reasoning tasks each turn to move the simulation forward. The first task is to generate potential action sets that are relevant to the current state. Using rules of the types (4) and (5), the system forward chains from the current state. Each rule can return a single action, or a set of related actions that must be taken together. Each of these singletons or sets is a dialogue action that the agent whose turn it is could choose to undertake.

The second task is to take those relevant actions and generate all the possible moves. Using rules of type (1), (2) and (5), the system abductively back chains from the target actions to the surface forms that, if used in the current context, would entail them. Each distinct surface form is added to a new potential move, along with the relevant actions that it entails. Surface forms that perform multiple relevant actions are noted as such, for possible preference in the user interface. Using the same rules, the system then forward chains from the surface forms to add all entailed action frames to each potential move. Finally, the surface frame must be rendered to natural language text. This is currently done using a simple template system, with verb choice reflecting aggressiveness. Aggressiveness and other action qualities that are in the move could also be displayed using icons, grouping, colors or any other such scheme.

The third task is to update the simulation once a move has been chosen by the agent whose turn it is. Given that all entailed actions are already present in the move, this process is simply forward chaining using rules of type (3).

The state condition antecedents and state change consequents in the various rule types can be either intermediate frames or anonymous, procedural functions. Intermediate frames are used in rules of type (5) that simply enable self chaining. Each instance of reasoning is always situated in a turn, and the game context allows us to assume that the prior simulation state, together with the prior moves, is both complete and unchanging. Thus a closed world assumption (Reiter 1978) can be applied, knowing that those procedural conditions will always evaluate to the same result for that turn.

Modularity

The simulation state definition, frame definitions and inference rules constitute a model of dialogue. However, such a model must integrate numerous concerns relevant to the setting, plot and characters involved. From both a user experience and AI research point of view, it would be greatly preferable to have a system that can be tested and extended in a modular fashion. The complete model is therefore exchanged for a federation of independent models, with special global models to hold them together. The system has been developed with class-based *modules*, each of which instantiates one of those narrower models. Each module holds a simulation state, a set of candidate action generation rules of type (4), a set of surface/action rules of types (1) and (2), a set of simulation update rules of type (3), and a set of simulation chaining rules of type (5). These rules are kept separate simply for efficiency in the three reasoning tasks.

In order to keep the modules independent, a small set of global models is defined. These models consist only of simulation state, frame definitions and inference rules between those frame types and other global model frame types. Global models do not generate potential actions or surface forms.

All reasoning takes place in either the global space or the context of a particular module. During global reasoning, the process must query all modules for any global frame. However, when reasoning within a module, the process only moves up to the global level when a global frame type is encountered. Thus, each module can only interact with global models, and the complexity of module-specific reasoning is not multiplied by the complexity of module-specific reasoning in every module in the system. Global models, on the other hand, potentially interact with each other and with every module. The effectiveness of this approach to controlling complexity depends entirely on the design of the global models. But it provides a choke point to keep every new module from exploding the complexity of the overall system.

Simulation Modules

Negotiation

The negotiation model does not explicitly represent two competing contracts moving together, as might be intuitively proposed. This is partly a design decision, recognizing that representing two separate contracts to the player in a game context would be a lot of information, and partly because, from a dialogue point of view, we want to focus on shared perception of discourse entities, including the contract. The two lists of obligations are stored as sets of frames, which can be reasoned over like any other set. As described above, each obligation has a scalar weight, which is equal in this prototype to the amount of gold payment that obligation is worth. The *balance* of the contract is the difference between the sums of the weights of obligations on each side. In reasoning, this balance is always characterized from the viewpoint of the agent whose turn it is, with positive numbers representing advantage to that agent. The balance in favor of an agent is his or her *advantage* while the balance against them is his or her *disadvantage*. Much of the reasoning about negotiation is done in terms of those advantages and disadvantages.

The negotiation state maintains the lists of obligations and the weight of each obligation. It also stores the current balance of the contract (in favor of the PC), and a flag to indicate whether the contract has been accepted. The state also stores the last offered advantage of each agent, which is the balance (in their favor, may be negative) of the last contract they proposed. These values start as null for both agents. Finally, the state includes a frame for details of the last offer that was made, storing the agent who made it, the obligation that was added or modified, the magnitude of concession and a flag indicating whether it was a partial offer. The magnitude of concession is the difference between the balance of that agent's prior last offered advantage and the balance of the last offered contract. A positive number indicates that the new contract diminishes their advantage (or increases their disadvantage). A partial offer indicates that the proposing agent did not intend for the proposed contract to be considered as-is, but is waiting for the other agent to fill in their side of the contract.

The rules for potential action generation entail dialogue action frames for the dialogue acts propose, counter and accept. Each frame has an attribute for *aggressiveness* which is set to *neutral*, *low*, *high* or *very high*. In the condition where the agent is at a contractual disadvantage, and there is no payment obligation already in the contract, three different actions proposing payment to the agent are generated, with the properties shown in Table 1. The fair value of a newly introduced payment is equal to the prior disadvantage, balancing the contract. Obviously, the particular percentages are subjectively chosen.

Proposed Payment Amount	Aggressiveness
Fair Value (FV)	Neutral
FV + 20% of contract	High
FV + 50% of contract	Very High

Table 1. Aggressiveness values for establishing initial payment in a disadvantaged condition.

In the case where the agent is at a disadvantage, and payment is already in the contract, and the agent has no last offered proposal, four actions countering with a modification of the payment obligation are generated, with the properties shown in Table 2. Mirroring is proposing an amount giving oneself the same advantage the prior contract gave the other agent. These rules can apply to the payer or the payee.

Proposed Payment Amount	Aggressiveness
Balance contract	Neutral
Mirror and decrease adv. 50%	Low
Mirror	High
Mirror and increase adv. 50%	Very High

Table 2. Aggressiveness values for altering payment in a disadvantaged no-prior-offer condition.

Assuming the prior contract put the other agent at a 20 point advantage, mirroring would propose a contract that put the acting agent at a 20 point advantage, setting up a "meet in the middle" at a balanced contract. More and less aggressive actions shift that middle. In addition, a type (1) rule specifies that, under the same conditions, proposing an amount closer to balanced than the mirror amount entails a *concession* action in that amount.

In the general case, where payment is already in the contract, and both agents have offered proposals, 3 actions countering with a modification of the payment obligation are generated, based on the whether the acting agent's last offered advantage is greater, less or equal to the current balance. Those conditions and action attributes are given in Table 3. Those more complex, these actions are still providing 3 levels of aggressiveness around qualitatively significant values in the range between the agent's last proposals. Split refers to the amount between the agent's last offer advantage and the current offer advantage to the other agent. Split/Mirror is the more advantageous of those two values to the acting agent. Unlike the prior conditions, the aggressiveness of these actions is based only on the magnitude of the proposed balance, as given in Table 4. Again, these rules can apply to the payer or the payee.

Agent's Last Offer Adv. (LOA) vs. Current Offer Adv.	**Proposed Payment Amount**
Equal	Balance contract
	Decrease LOA 50%
	Hold (return to last offer)
Greater Than	Lesser of Split/Mirror
	Greater of Split/Mirror (GSM)
	Decrease LOA 50% of GSM
Less Than	Decrease LOA 50%
	Balance contract
	Decrease LOA 50% of Split

Table 3. Generated proposal values for altering payment based on prior offers.

Proposed Advantage	**Aggressiveness**
0	Neutral
0 to 10% of contract	Low
10 to 20% of contract	High
> 20% of contract	Very High

Table 4. Aggressiveness values for altering payment based on prior offers.

In addition, type (1) rules specify that if the last offer from the other agent included a concession, then if this offer includes a concession in at least that amount, it entails a *match* action. If the current concession amount is higher than the prior, then it also entails a new *concession* action for the amount of the difference. If it does not at least match, then it entails a *match-reject* action.

Finally, if the contract is balanced or tilted in favor of the acting agent, and the other agent made the last offer, then an accept action frame is generated. Accepting is always neutral aggression.

Surface rules in the negotiation model simply map from the propose, counter and accept frames to the surface forms shown in the example dialogue. Simulation update rules are also quite straightforward, with proposal and counter actions adding new obligations and modifying the parameters of existing ones. The last offered advantages and last offer details are also updated. A partial proposal updates the last offer details, but does not assign the agent a last offered advantage. Accepting a contract ends the dialogue.

Alliance and Reputation

The global model of *alliance* is concerned with how characters perceive each other being "on their side". The simulation of this model maintains such an estimate for each character towards each other character as a scalar value. The model defines frames for actions that strengthen and weaken alliance. Simulation update rules apply those changes with diminishing returns as the estimation becomes very positive or very negative.

Unlike adding independent modules, adding a new global model to the system requires altering the existing modules. In the case of the negotiation module, actions with the aggressiveness attributes of *low, high*, and *very high* entail increasingly strong weakening of alliance from the patient to the agent of the action. In similar fashion, concession and match actions entail strengthening of alliance while match-reject actions entail weakening. An additional rule, to test the inter-module reasoning, specifies that offering a balanced contract when holding an advantage (the agent's last offer advantage is greater than the current advantage for the other agent) also entails strengthening of alliance. This action is not generated by the negotiation module as a potential choice.

The *reputation* module simulates the reputation of the PC with the group represented by the NPC, as described in the hypothetical game. The inference rules for reputation state that weakening and strengthening the alliance from the NPC to the PC also weakens and strengthens the reputation. The only difference in the two values is in their initial values – the group may already have an opinion of the PC while the NPC does not. At this time, this module is only there as a module interaction test case. The reputation module generates a single potential action in all situations, calling for increase of reputation. When the system attempts to abduce surface forms for this potential action, it reasons that reputation can be increased by strengthening alliance, which can be strengthened by, among other things, offering a balanced contract while holding an advantage. By bridging through the global model, the reasoning process is able to abduce the surface form of offering a contract (in the right situation) to meet the reputation goal. This adds another choice that the negotiation module would not have generated.

Discussion and Future Work

This prototype exploration is focused on a small segment of negotiation dialogues, within the constraints set out by our hypothetical game context. Nevertheless, it has contributed a promising framework for the difficult challenge of modular integration of the wide range of models that are relevant to simulating dialogue. This framework has helped refine the proposed game context, which we believe will enable productive investigation of dialogue in interactive narrative from both the user experience and artificial intelligence perspectives. The modular design also provides a clear path for incremental addition of models. We have already begun work on argumentation, to allow agents to make and challenge claims about the value weights of various obligations, shifting the balance of negotiation during the process. We have also started on social positioning, the so-called *shadow negotiation* (Kolb 2004), where agents fight to increase and maintain personal credibility during negotiation. Following that, our next concern will be with emotional reactions and personality in interpretations. Once we have larger, more robust modules in place, we can turn to more rigorous evaluation of the reasoning framework and modular design. The last two modules in particular will require a more robust NL generation approach, and would benefit greatly from integration with emotive avatars. The PERSONAGE system (Mairesse and Walker 2007) performs natural language generation taking personality into account, which is a direction we would like to explore.

We are also in the process of embedding this dialogue system in a simple video game for testing user experience. The most pressing question is whether this type of dialogue simulation, with the goals and constraints we have placed on it, can communicate the narrative concerns of setting, plot and character to the player. This is a key question, as dialogue in general, and traditional quest text in particular, is included in most games solely for that purpose.

At this point, we have not yet approached the problem of strategic, consistent NPC choice selection. There are numerous approaches to this problem, including intention-based hierarchical planning (Riedl and Young 2010) and application of character archetypes (Rowe et al. 2008). The inclusion of consistent characterization, through personality or attitude, is of particular interest.

Acknowledgements

This project was supported in part by the UTPA Center of Excellence in Science Technology and Mathematics Education Undergraduate Research Program, funded by Office of Naval Research contract W911NF-11-1-0150.

References

Marc Cavazza, Fred Charles. 2005. Dialogue Generation in Character-based Interactive Storytelling. In *Proceedings of the First Annual Artificial Intelligence and Interactive Digital Entertainment Conference*, Marina del Rey, California, USA.

Costikyan, G. 2007. Games, Storytelling, and Breaking the String. *Second Person: Role-Playing and Story in Games and Playable Media*. Cambridge, MA: MIT Press.

Knickmeyer, R. and Mateas, M. 2005. Preliminary Evaluation of the Interactive Drama Façade. In *CHI '05 extended abstracts on Human factors in computing systems*.

Kolb, D. 2004. Staying in the Game or Changing It: An Analysis of Moves and Turns in Negotiation. *Negotiation Journal* (April).

Koster, R. 2004. *A Theory of Fun for Game Design.* Paraglyph Press.

Loyall, A. B., Bates, J. 1997. Personality-Rich Believable Agents That Use Language. In *Proceedings of the First International Conference on Autonomous Agents*. pp. 106-113.

Jonathan Rowe, Eun Young Ha, and James Lester. 2008. Archetype-Driven Character Dialogue Generation for Interactive Narrative. In *Proceedings of the Eighth International Conference on Intelligent Virtual Agents*, Tokyo, Japan, pp. 45-58.

Mairesse, F., Walker, M. 2007. Personage: Personality generation for dialogue. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*.

Mateas, M., and Stern, A., 2002. A Behavior Language for Storybased Believable Agents. In *IEEE Intelligent Systems, special issue on AI and Interactive Entertainment*.

McCoy, J., M. Treanor, B. Samuel, B. Tearse, M. Mateas, and N. Wardrip-Fruin. 2010. Comme il Faut 2: a fully realized model for socially-oriented gameplay. *Proceedings of the Intelligent Narrative Technologies III Workshop*, New York, NY, USA, ACM, pp. 1–8.

Mehta, M., Dow, S., Mateas, M., MacIntyre, B. 2007. Evaluating a Conversation-Centered Interactive Drama. In *Proceedings of the Sixth International Conference on Autonomous Agents and Multiagent Systems*.

Riedl, M. and Young, R.M. 2010. Narrative Planning: Balancing Plot and Character. *Journal of Artificial Intelligence Research* 39 (2010) 217-2.

R. Reiter. 1978. On closed world data bases. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pp. 119–40. Plenum Publ. Co., New York.

Jonathan Rowe, Eunyoung Ha, James Lester. Archetype-Driven Character Dialogue Generation for Interactive Narrative. 2008. In *Proceedings of the Eighth International Conference on Intelligent Virtual Agents* (IVA-2008), Tokyo, Japan.

Sali, S., Wardrip-Fruin, N., Dow, S., Mateas, M., Kurniawan, S., Reed, A., Liu, R. 2010. Playing with Words: From intuition to evaluation of game dialogue interfaces. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*.

Traum, D, Swartout, W., Gratch, J., Marsella, S. A Virtual Human Dialogue Model for Non-Team Interactions. 2008. *Text, Speech and Language Technology*, Volume 39.