Digital Improvisational Theatre: *Party Quirks*

Brian Magerko, Christopher DeLeon, and Peter Dohogne

Georgia Institute of Technology, Technology Square Research Building,
85 Fifth Street NW, Atlanta, Georgia 30308 United States
{magerko,cdeleon3,pdohogne3}@gatech.edu

Abstract. This paper describes the creation of a digital improvisational theatre game, called *Party Quirks*, that allows a human user to improvise a scene with synthetic actors according to the rules of the real-world version of the game. The AI actor behaviors are based on our study of communication strategies between real-life actors on stage and the fuzzy concepts that they employ to define and portray characters. Development of content for the system involved the creation of a novel system for animation authoring, design for efficient data reuse, and a work flow centered on parallel data entry and rapid iteration. A subsequent user test of the current system is presented as an initial evaluation of the user-centered experience in participating in a virtual *Party Quirks* game.

Keywords: Improvisation, Theater, Player Experience, Virtual Characters.

1 Introduction

Playable improvisational theatre games are a form of game-based entertainment that has rarely been tackled within the game AI community [1], [2]. Improvisational AI characters have been developed since the early to mid-1990's [2-6], though none of these approaches were informed by an cognitively-based understanding of how improvisers communicate and construct stories together. Our study of improv actors, both in lab settings and in real performance settings, has informed a grounded theory-based (i.e. data-driven instead of based on pre-conceived hypotheses) analysis to inform the construction of simple AI agents (removed for blind review). The construction of these agents has cyclically informed our data analysis process by highlighting what data we need more of and what fallacies exist in the current analysis. This has enabled a richer data analysis and, in turn, more complex agents to be built.

This work in studying improvisational theatre has resulted in the implementation of the real-time improve performance game *Party Quirks*, which is populated by improvisational synthetic actors and intended to be an example of certain aspects of our findings as a middle computational step rather than as a final, complete improvisational system. *Party Quirks* was emulated due to its focus on character portrayal and lack of emphasis on story construction. Story construction is a component in many improvisational games but an overly complicated problem for our initial system and is a current focus of our theoretical work [7], [8]. A typical game of *Party Quirks* involves four players: one plays the role of party host, and all others play as party guests. Each party guest is assigned a "quirk" – some special trait for each guest that

H. Högni Vilhjálmsson et al. (Eds.): IVA 2011, LNAI 6895, pp. 42–47, 2011.

is public knowledge to everyone except the party host, including the audience. The host player then, within the context of hosting a party, aims to figure out what quirk each guest is portraying through their interactions. A guest typically leaves the scene when the host has successfully guessed their quirk.

Our digital version of *Party Quirks* consists of software agents acting independently to emulate the communication processes and reasoning about ambiguity that live actors demonstrated during performances in our empirical studies. A human plays as the host in the virtual scene. As opposed to relying on behind-the-scenes communication, the agents have to rely on human-like methods of communication on stage because they are improvising with a human in the scene as an equal. We call this an *equal mixed-initiative* theatre game.

The following sections in this paper provide a brief overview of the knowledge model used by our synthetic actors followed by a description of the *Party Quirks* implementation and evaluated user experience during a live demo event.

2 Knowledge Model

We selected a set of 18 basic *character prototypes* (e.g. *Pirate* and *Cowboy*) as possible quirks to make the content authoring tractable but non-trivial. A *prototype* refers to an idealized socially recognizable constructs that map to a certain kind of character [9]. We define each prototype as a collection of properties with varying degrees of membership (DOM) in sets that represent character attributes. This approach is similar to how we have seen portrayals of prototypes in our human data and matches well to contemporary thoughts on how humans categorize fuzzy concepts [9], [10].

Attributes are adjectives that define a prototype. *Actions* are the physical acts that are used to communicate attributes and are associated with at least one <attribute, DOM range> pair. For example, <*uses_magic*, 0.8-1.0> implies a high association with magic usage, which is connected to the action *controlWeather*. Any character with *uses_magic* between these values can therefore execute the *controlWeather* action on stage.

The primary benefit of using fuzzy membership of sets is that it captures the ambiguity inherent in stage performance. Performing an action with multiple interpretations can lead other actors to have different interpretations of the action than were intended, which often happens in performances. The calculated ambiguity values also provide the means to determine how much the host's interactions indicate their convergence with the "reality" of the scene. In other words, the actions that a human host executes indicate how close they are to guessing a guest's quirk (i.e., reaching cognitive consensus [11]).

3 Experience, Implementation, and Testing

The interaction model for our *Party Quirks* system was modeled after the observed rules that human actors use in real life games of *Party Quirks* (anonymous). The user, as the party host who has to guess each guest's quirk, inputs commands into an iPad. The interface consists of textual buttons that are divided into branching options to reduce the amount of information on-screen at once. The choices given to the

user/host are based on the processes that improvisers use to negotiate shared mental models (i.e. shared understandings of the details of a scene as a scene progresses) on stage during earlier phases of this study [11]. The implementation presented here is based on these findings.

The *Party Quirks* actors and stage are projected onto a screen at near life-size scale in an attempt to help immerse the user as one of the four improvisational actors. A webcam stream of the user is placed next to the stage as a means of representing the user in the virtual space and as a location to place a speech bubble that displays user actions.

The user makes menu selections from an iPad interface that represents the different abstract communication moves observed in our human data [11]. The iPad was utilized as an input device for several reasons. A controller was needed which would allow the user to stand, since all guest actors are presented as standing. A touchscreen enables buttons to be dynamically labeled, reducing the complexity of the interface by breaking interactions into categories and allowing for a naturalistic input modality (i.e. finger tapping). As the user makes menu selections, including doing nothing, the AI actors respond on the virtual stage via animations and dialogue portrayals.

3.1 Technology

The AI model for character behaviors was implemented in Processing to simplify cross-platform development and afford rapid prototyping. All data is obtained from Google Docs spreadsheets at program start up via the Google Docs API, enabling parallel online authoring and immediate use by the application without recompiling. The spreadsheets contain definitions of prototypes, attributes, actions, as well as the degrees of membership between prototypes and attributes, and between actions and attributes. The spreadsheets also include text-based dialogue utterances that are associated with the actions.

The animation system was also built using Processing, enabling animation playback to be directly integrated with the AI model. Animations exist as a series of skeletal poses, which play through in sequence. The poses are stored as a list of angles and percentages for each limb, indicating its rotation at the base joint and its scaling lengthwise (used, for example, to point at the camera), as well as three integers corresponding to an enumeration of facial expressions and hand poses. Each animation is saved as a separate file to facilitate parallel development without needing to merge data.

3.2 User Testing

Notes were taken on user experience issues during a public showcase in 2010 and users were encouraged to share feedback. The development team used this feedback to identify sources of frustration and confusion, exposing incorrect assumptions made during the design.

Users found difficulty dividing attention between the iPad and the projector screen. The virtual actors offered information, in the form of animations, while the user was still busy trying to read and interpret options on the iPad. Confirmation screens, displayed on the iPad at the end of each input sequence, turned out to be unexpected,

leading the user to watch the projected image in anticipation of response while the device in hand prompted silently for one last action. We anticipated split-attention as a potential problem when designing the system, but opted for trying it in lieu of more computationally difficult interfaces (e.g. natural language interaction via spoken word) or pragmatically difficult ones (e.g. tangible items used to represent the myriad of options present in the iPad user interface). In the interactive examples mentioned in the introduction [4], drop-down menus gave the user a choice among potential goals and behaviors. The only similar system to attempt full natural language interaction, Mateas and Stern's *Façade*, has had mixed success due to the difficulty in conversing with synthetic characters in a broad conversation domain [12]. This motivated the team's decision to scope the work to focus on the myriad issues that inherently come with trying to build an AI-based digital improv system and to avoid using full natural language interaction for this installation. Future work will examine more natural interaction modalities.

Users also found difficulty splitting attention between the three actors simultaneously. This helped illustrate what may be a difference between having trained improv actors as users instead of untrained attendees at a university media showcase. Those players that played more than one round fared much better on attempts after the first, demonstrating that with prior exposure to the structure the game could be played more successfully. This discovery led the recent development of a single-guest tutorial round for first-time users.

One of the simplest strategies to gather information from the guests about their quirks – a strategy that occurs in live improv and is practical in this software implementation – is to guess prototypes even before guess confidence is high. This narrows down the potential prototype answers by prompting new contrasting information from the guests. However, many simulation players seemed reluctant to make prototype guesses until they were confident in their answer, possibly from confusion over whether some penalty might be imposed for incorrect guesses or simply due to a lack of experience in improvisational acting.

In some cases, there was ambiguity in what the middle value should mean for prototype / attribute degree of membership values. For example, if "explaining relativity" signifies high intelligence, "reading a book" might suggest comparatively normal intelligence, although a player might interpret book reading as a sign of high intelligence. Although different interpretations of values between extremes were a source of confusion, this type of confusion is a normal part of *Party Quirks*; different people have different models of norms and extremes in the real world. These misunderstandings can occur between two live actors just as they can between the data set's author and a human player. Future work will involve gathering crowdsourced data to provide DOM distributions based on a much larger set of authors to define prototypes (as opposed to just the research team's intuitions).

4 Discussion

While we are encouraged by the initial work done in *Party Quirks* in representing character prototypes, the process of building shared mental models, and an initial communication framework for interacting with improvisational characters, this initial

system is not without its drawbacks. Users often get stuck just guessing repeatedly instead of making use of the other moves common in performances. This points to a major issue of presence in the system; users do not act like they are performing with the actors on a virtual stage, but like they are prodding a system to see how it responds. The virtual actors give an often-entertaining response with any guess, which provokes the user to guess again instead of selecting other moves. Future work in interface design, such as using voice commands or gesture recognition, may help actively involve the user in the performance space rather than acting outside of it and getting stuck in the most convenient menu option.

The agents themselves are fairly generalizable in terms of the number of attributes that can be used to describe characters and the different mappings from attribute value ranges to actions that can occur. However, they cannot be altered, augmented, or combined. For instance, prototypes cannot be blended together to create new prototypes (e.g. a mosquito that acts like a drunk when it drinks blood) nor can they be created with some antithetical property (e.g. a plumber who is afraid of water). This points to the need for future work to focus on how agents can employ the process of *conceptual blending* [13].

Another major limitation of the agents is that they have no concept of narrative; they are incapable of constructing a story or having dialogue acts that logically progress over time. This issue has fueled our current research agenda of exploring conceptual models of equal mixed-initiative (i.e. AI and humans are both on the virtual stage and equally share responsibility in constructing the story) collaborative story construction from the viewpoint of a) setting up the details of a scene (e.g. where the scene takes place, who the characters are, what joint activity they are doing together, etc.) and b) finding the "tilt" for the scene (i.e. what the main dramatic focus of the scene is)[14]. The future of this work will be a synthesis of these lessons learned from *Party Quirks*, resulting in a troupe of AI improvisers than can jointly construct narrative on stage with or without a human acting with them.

References

[1] Hayes-Roth, B., Brownston, L., van Gent, R.: Multiagent collaboration in directed improvisation. In: Proceedings of the First International Conference on Multi-Agent Systems (ICMAS 1995), pp. 148–154 (1995)

[2] Bruce, A., Knight, J., Listopad, S., Magerko, B., Nourbakhsh, I.R.: Robot Improv: Using Drama to Create Believable Agents. In: AAAI Workshop Technical Report WS 1999-15 of the 8th Mobile Robot Competition and Exhbition, vol. 4, pp. 4002–4008 (1999)

[3] Perlin, K., Goldberg, A.: Improv: A System for Scripting Interactive Actors in Virtual Worlds. In: SIGGRAPH 1996, New Orleans, LA (1996)

[4] Hayes-Roth, B., Van Gent, R.: Story-Making with Improvisational Puppets and Actors. Technical Report KSL 1996-09, Palo Alto, CA (1996)

[5] Swartjes, I., Theune, M.: An Experiment in Improvised Interactive Drama. Intelligent Technologies for Interactive Entertainment, 234–239 (2009)

[6] Harger, B.: Project Improv, Project Improv (2008), http://www.etc.cmu.edu/projects/improv/ (accessed February 25, 2010)

[7] Baumer, A., Magerko, B.: An Analysis of Narrative Moves in Improvisational Theatre. Presented at the International Conference on Interactive Digital Storytelling, Edinburgh, Scotland (2010)
[8] Baumer, A., Magerko, B.: Narrative Development in Improvisational Theatre. Presented at the International Conference on Interactive Digital Storytelling, Guimarães, Portugal, pp. 140–151 (2009)
[9] Lakoff, G.: Cognitive models and prototype theory. In: Margolis, E., Laurence, S. (eds.) Concepts and Conceptual Development: Ecological and Intellectual Factors in Categorization, pp. 63–100 (1987)
[10] Rosch, E.: Principles of categorization. In: Margolis, E., Laurence, S. (eds.) Concepts: Core Readings, pp. 189–206 (1999)
[11] Fuller, D., Magerko, B.: Shared mental models in improvisational performance. In: Proceedings of the Intelligent Narrative Technologies III Workshop, Monterey, CA (2010)
[12] Mateas, M., Stern, A.: A Behavior Language for Story-Based Believable Agents. IEEE Intelligent Systems 17(4), 39–47 (2002)
[13] Fauconnier, G., Turner, M.: The Way We Think: Conceptual Blending and the Mind's Hidden Complexities. Basic Books, New York (2003)
[14] Brisson, A., Magerko, B., Paiva, A.: Tilt Riders: Improvisational Agents Who Know What the Scene is About. In: 11th International Conference on Intelligent Virtual Agents, Reykjavik, Iceland (2011)