Story-Making with Improvisational Puppets and Actors

Barbara Hayes-Roth and Robert van Gent[1]

Computer Science Department

Stanford University

December, 1995

[1]The work reported in this paper was supported by: ARPA Contract N66001-95-D-8642-Subcontract #137-1 through Teknowledge, Inc., a gift from Intel, and Seed Grants from Stanford's Center for the Study of Language and Information and its Office of Technology Licensing. The work has benefited from discussions with present and past members of the Virtual Theater research group, including: Lee Brownston, Erik Sincoff, Ruth Huard, Todd Feldman, Jack Jen-Jay Yu, Katherine Isbister, Daniel Rousseau, and Brian Lent. It also has benefited from discussions with Stanford Professors Larry Friedlander (English Literature) and Patricia Ryan (Drama). We thank Joseph Bates of CMU and Ken Perlin of NYU for allowing and assisting us in the use of their animation applications. We thank Aaron and Nora Hayes-Roth for creating the voices, lines, and personalities of two animated puppets. A patent application has been filed for the "Method and System of Directed Improvisation by Computer Characters."

Abstract

Improvisational actors create engaging vignettes in real time, without detailed planning, and often working within constraints provided by the audience. We are exploring the possibility of creating intelligent computer agents that can be embodied as animated characters, can perform in a manner loosely resembling that of human improvisors, and can tailor their performances to abstract directions offered by users or other system components. We describe implemented systems in two paradigms. Improv Puppets improvise under players' directions in real time. Improv Actors improvise under scenarios provided in advance. We discuss potential applications related to entertainment, the arts, and children's learning toys.

1. Directed Improvisation

Improvisational actors create their performances in real time, without detailed planning, and often working within constraints provided by the audience. Because improvisational performances are spontaneous, they do not reach for the artistic heights of conventional theater. Nonetheless, audiences find them entertaining. A skilled improv troupe produces stories that are pleasing in form and content. Even when they are only moderately successful, it is amusing to observe the actors work to create a story while meeting the constraints. There is a special pleasure in knowing that each performance is unique.

We are exploring the possibility of creating intelligent computer agents that can be embodied as animated characters, can perform in a manner loosely resembling that of human improvisors, and can tailor their performances to abstract directions offered by users or other system components.

To illustrate the baisc idea, Figure 1 presents a hypothetical episode involving two agents embodied as animated characters: a large character (Tory) and a small character (Scout). Tory has been directed to act curious and friendly. Scout has been directed to act playful. These directions are quite general; neither character has been directed to do anything in particular, only to behave in ways that reflect the specified moods. Each character incorporates the directions into the "control plan" that will guide his or her own behavior (0) for the duration of the episode.

Working within the constraints of the directions, each character improvises his or her own behavior, while simultaneously interpreting and responding to the partner's behavior as the episode unfolds. At first, Tory is alone in the world. He has many possible behaviors, but he can't act friendly without a partner. Following his direction to act curious, Tory decides to look around for something (1). Scout enters (2), observes Tory standing still, tries to interpret his behavior, but infers nothing (3). Following her direction to act playful, she decides to start something: to play alone for a while and then hide (4). Observing Scout enter and begin to play alone, Tory tries to interpret her behavior and infers that she is shy (5). Following his direction to act friendly, specialized for a shy character, Tory decides to approach Scout, greet her, and invite her to play (6).

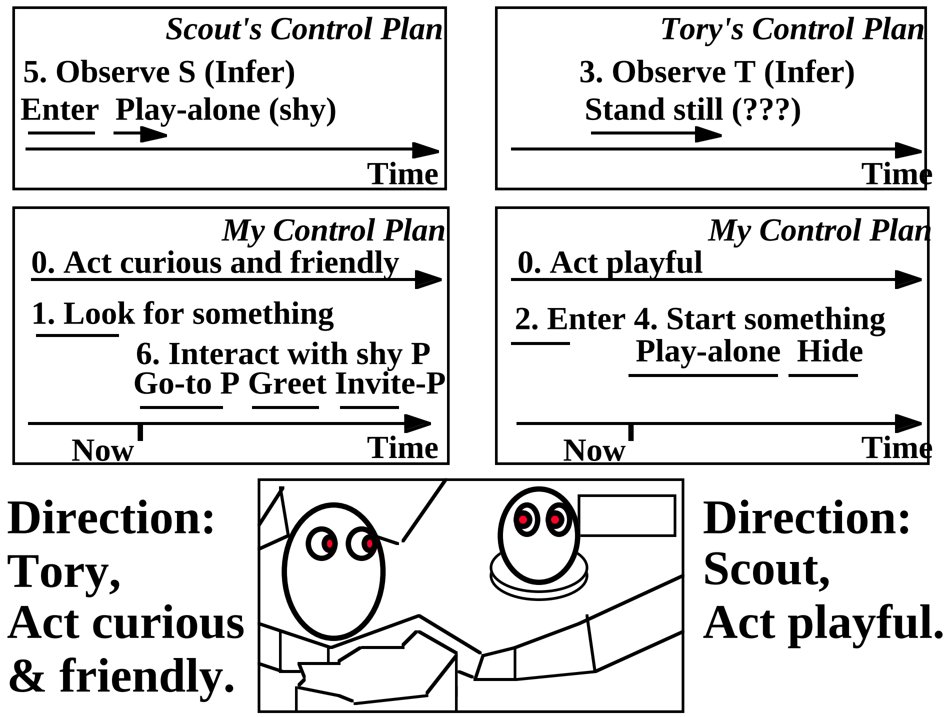

Figure 1. Two characters improvise within the constraints of their directions. The box immediately above each character shows its "control plan" for its own behavior. The box above that one shows the control plan a character attributes to its partner, based on abductive inference from the partner's behavior. Each element in a control plan, numbered 0-6 in order of generation, indicates an "intended" kind of behavior. (The characters shown are embodied as "Woggles," after the animation developed by J. Bates of CMU.)

Although they don't know it, Tory and Scout now have incorrect models of one another's personalities and behavior and conflicting plans for the interaction. One of them must change, but it doesn't really matter which one. If Tory invites Scout to play first, Scout will drop her own plan and accept his invitation. If

Scout hides first, Tory will drop his plan and join her game. Following the most basic rule of improvisation, the characters *accept all offers* [24]. Each one actively seeks to interpret the other's behavior and readily changes his or her own plans to accommodate the other's apparent intentions.

More generally, we envision these capabilities. Each agent, embodied as an animated character, would accept directions from one or more exogenous sources, either in real time time or in advance of a performance. The directions would constrain, but not completely specify the agent's behavior. For example, they might assign the agent a role, endow it with personality features, change its mood, or instruct it to perform a kind of behavior. A sequence of directions might shape the narrative structure of the agent's individual behavior and interactions with other agents. Each agent would construct its own detailed course of behavior, following directions, interacting appropriately with other agents, and filling in unspecified elements along the way. Its improvisations would make sense in the real-time situation, reflect its role, personality, and mood, and manifest certain life-like qualities, such as normal variability and idiosyncrasies in behavior. In some cases, the agent's improvisations might contribute to the narrative form and content of the story. If a given sequence of directions were repeated on different occasions, the agent might improvise different performances, surprising the audience and perhaps even its director.

We believe that improvisational agents, possessing the kinds of capabilities discussed above, would be useful components in a variety of applications, especially those related to entertainment, the arts, and children's learning toys [4,]. Applications can be framed in different interaction paradigms, including two we have been studying, "Improv Puppets" and "Improv Actors." In both paradigms, users and agents collaborate on the creation of simple "stories" performed by the embodied agents. However, users direct Improv Puppets interactively in real time, while they direct Improv Actors with scenarios constructed in advance. As discussed below, these differences also affect other properties of the collaboration and therefore support different kinds of applications.

Section 2 describes our technical approach to creating improvisational agents. Sections 3 and 4 describe two implemented systems: "Animated Puppets," illustrating the Improv Puppets paradigm; and "Master and Servant," illustrating the Improv Actors paradigm. (Our conference presentation will include videotaped demonstrations of both systems.) Section 5 draws conclusions.

2. Technical Approach

2.1 Logical System Organization

An agent has a "body," a "mind," and a "mind-body interface" (see Figure 2).

An agent's body is a computer program controlling a graphical manifestation in a virtual world, which it may cohabit with other agents' bodies. It has sensors and effectors to perceive events and execute motor behaviors in the virtual world. An agent also may have an interface for communicating with users.

An agent's mind is a computer program that performs three basic functions. It integrates perceptual inputs withknowledge and inferences to assess the agent's dynamic situation. It instantiates and decides when to execute particular behaviors. It performs all processing intervening between situation assessment and behavior. The architecture of the mind is discussed below.

An agent's "mind-body interface" mediates interactions between its mind and body in a tight real-time control loop. It classifies patterns of sensor data as meaningful perceptions, which it relays to the mind. It translates motor commands from the mind into sequences of executable instructions, which it relays to effectors. Thus, the mind-body interface allows an agent's mind to be transferred among different bodies without modification and vice versa, so long as the agent is given an appropriate set of mappings between sensor data and perceptions and between motor commands and effector instructions.

Figure 2 illustrates our logical system organization of these components (see also [9]). This organization permits alternative assignments of system components to processors and to physical locations, as illustrated below.

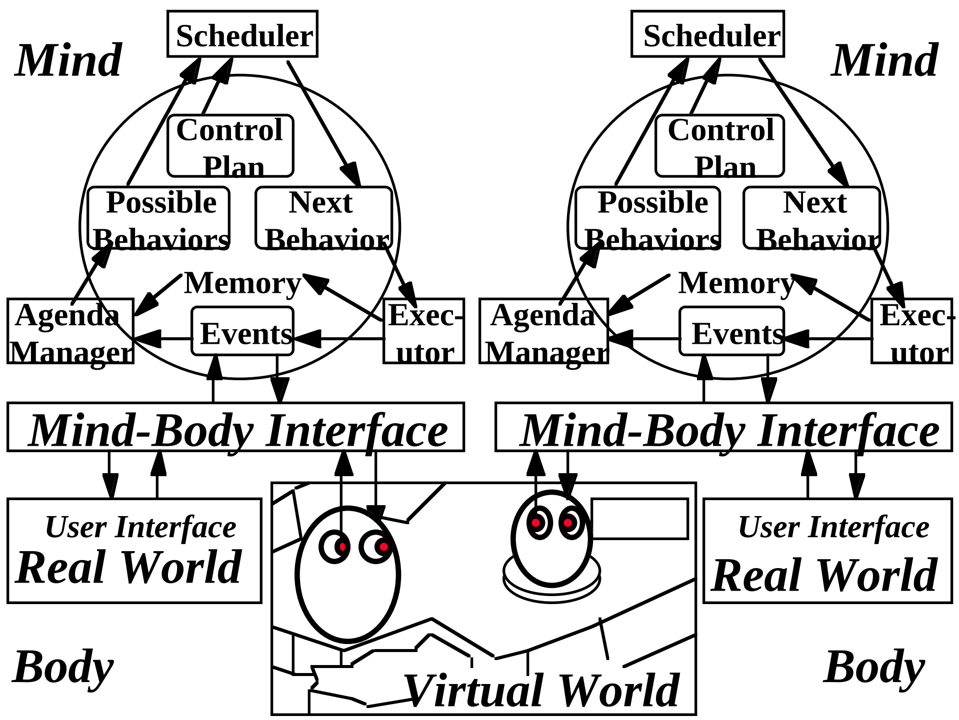

Figure 2. Logical system organization for two agents in a virtual world.

2.2 Architecture of the Mind

Each agent's mind instantiates and elaborates a *dynamic control architecture* , which has been applied in other intelligent agent applications [5, 7, 9] and is available as the BB1 software system. As illustrated in Figure 2, the architecture iterates a three-step execution cycle. First, an *agenda manager* uses recent perceptual and events, state information, and knowledge to instantiate behaviors that are *possible and relevant* in the present situation. Second, a *scheduler* selects relevant behaviors that satisfy the constraints in the current *control plan.* Third, an *executor* sends the associated behavior commands to the mind-body interface. It also executes associated cognitive behaviors, which produce cognitive events and may cause changes in state, including the contents of control plans.

The agent architecture can exploit many different kinds of knowledge and information in various data structures. Here we characterize the knowledge structures used for agents in the two experimental applications discussed below.

Potential behaviors are organized in a class hierarchy. Each behavior has a set of relevance conditions that can be satisfied by the occurrence of certain classes of events. When a behavior's relevance conditions are satisfied, the agenda manager instantiates the behavior for the current situation. The class hierarchy allows the agenda manager to search efficiently for relevant behaviors.

Each behavior has annotations specifying "typical" values on a set of mood variables (e.g., happy, sad). A behavior is more appropriate when the agent's actual mood approximates that behavior's typical mood values. The agenda manager uses mood values and some random variability to prune the current set of relevant behaviors. The scheduler uses mood values and some other information to choose which relevant behaviors to execute.

Each leaf in the behavior hierarchy has an execution script, which the executor instantiates and executes when that behavior is scheduled. Scripts are represented in a simple, restricted programming language that permits variable instantiation and conditional logic, terminating in commands to the agent's body.

Agents maintain several kinds of state. Mood values are set by directions from players (or other sources) or computed as functions on any event and state variables. State machines track agents' interactions. The locations and behaviors of static and dynamic entities in the virtual world, including each of the agents, are also represented explcitly. Directions are represented as constraints.

3. Improv Puppets

3.1 Improv Puppets Paradigm

"Improv Puppets" is an interaction paradigm modeled after traditional puppet play. With physical puppets, children move their respective puppets and speak the puppets' words in real time. Thus, the puppets serve as vehicles for improvisational storymaking by the children. Similarly, with Improv Puppets, players direct the behavior of their respective puppets in real time. Now, however, the puppets are animated, smart, and collaborative. Improvisational storymaking is a real-time collaboration among players and puppets. Below we describe a particular Improv Puppets system, called "Animated Puppets."

3.2 "Animated Puppets" Implementation

The logical system organization for Animated Puppets is similar to the one in Figure 2, with system components distributed among processes, platforms, and locations. Each puppet's mind (including its mind-body interface) and its user interface run as separate processes on a single Sun SPARCStation 10 in our laboratory. Multiple copies of the virtual world, each one controlling all of the puppets' bodies and displaying one puppet's user interface (see Figure 3), run on multiple SGIs in other buildings at Stanford. The puppets' mind-body interfaces broadcast instructions to their bodies in all copies of the virtual world. Players direct their own puppets and view the joint performance at individual SGI's.

The puppets have animated bodies, children's voices, and real-time user interfaces. For bodies, we adapted the "Woggles" animation developed by Joseph Bates at Carnegie Mellon University (see Figure 3). Augmenting their basic physical behaviors (e.g., leap to x,y, stretch, turn, blink), we created a few new gaits, so that puppets can introduce normal variability and mood-related expressiveness into their movements. For voices, we gave each puppet 250 lines, conceived and recorded by Aaron and Nora Hayes-Roth, ages 14 and 11, in response to elicitation cues comprising classes (e.g., greeting, invitation to play) and qualitative values of three moods: happy-sad, peppy-tired, friendly-shy. Thus, the puppets can engage in a variety of interactions and introduce normal variability and mood-related expressiveness into their verbal behavior. The children's voices and words endow them with personalities. The user interface has buttons, sliders, and a clickable world (see Figure 3), as discussed below.

The puppets' minds instantiate the general architecture described above, but introduce two additional steps into the control loop:

(a) Whenever the mind identifies a new set of relevant behaviors or changes its mood, it sends corresponding new instructions to the user interface.

(b) Whenever the mind receives a new direction from the user interface, it improvises a course of behavior consistent with that direction.

Each puppet has class hierarchies for 15 physical and 250 verbal behaviors. For example, "hop to destination" is a sub-class of "go to destination;" "greet friend" is a sub-class of "speak to friend." Each behavior has a set of relevance conditions and typical mood values (discussed below). Leaf behaviors have execution scripts that can be instantiated, interpreted, and executed. Simple scripts (e.g., hope to destination) directly command particular instruction sequences to the animated bodies. Complex scripts (e.g., Play alone for awhile) recursively invoke sequences of instantiated behaviors from the hierarchy. Script parameters permit different improvisations.

Although the puppets' class hierarchies are identical, their executable behaviors are different. Physical behaviors are different because of sizeappropriate differences in animation parameters. Verbal behaviors are different because of differences in the children's voices, intonations, and wordings.

Each puppet has three bipolar continuous mood dimensions: an emotional dimension (happy-sad), a physiological dimension (peppy-tired), and a social dimension (friendly-shy). Players can direct puppets' moods, but moods also change automatically when certain kinds of events occur. For example, a puppet will be happier if its invitation to play is accepted and sadder if its invitation is declined. A puppet will get tired if it plays hard and will pep up if it sleeps.

Each puppet has a state machine to track interactions with the other puppet (e.g., greeting exchanges, how-are-you exchanges, game-playing interactions). The puppets know properties of thirty locations in the virtual world. Each puppet tracks its own and the other puppet's location and behaviors.

3.3 Story-Making with Improv Puppets

Players and Improv Puppets collaborate to improvise stories in real time. Players collaborate with one another as they do when playing with physical puppets—but now they direct their puppets' moods, actions, and words. Puppets collaborate with one another by perceiving and responding to one another's behavior. Each puppet's behavior might elicit an automatic response from the other, change its mood, or cause it to identify new relevant behaviors. Puppets and players collaborate to determine the puppets' moods and behaviors. As illustrated below for the "Animated Puppets" system, this collaboration occurs as a mixed-initiative interaction through a real-time user interface (see Figure 3): puppets offer directorial options; players choose among the available directions; and puppets improvise under the chosen directions.

Each Animated Puppet offers its player a dynamic set of directorial options corresponding to a small, context-relevant subset of its full behavioral repertoire.

Treating its user interface as a “window” into its mind, a puppet reveals how it feels—emotionally, physically, and socially—in the positions of its sliders. It reveals what it is thinking about doing and saying as active blue and red buttons. It reveals where it is thinking about going as active areas of the virtual world. All of these “thoughts” and the corresponding interface displays change with the puppet’s situation. For example, a puppet often thinks about going somewhere—but where and how it wants to go depend on its mood. A tired puppet will consider only nearby destinations and calm gaits, while a peppy puppet will consider distant destinations and energetic gaits. On the other hand, travelling to a distant destination with a peppy gait will tire a puppet more than travelling to a nearby destination with a calm gait. As a second example, a puppet that is playing alone ordinarily will want to continue, offering the player only one directorial option: to stop playing. However, if the other puppet invites it to play, the puppet will offer its player new directorial options to accept or decline.

Each player directs his or her puppet by choosing among current directorial options. He or she can move the sliders to change the puppet’s mood, select a location in the virtual world to direct the puppet toward that destination, or select a blue or red button to direct the puppet to do or say something.

Each puppet immediately obeys its player’s directions by improvising an appropriate course of behavior, coloring its improvisations with life-like qualities: normal variability, idiosyncrasies, mood-related modulations of behavior, event-based changes in mood, and adherence to social conventions.

Taking a simple example, a puppet will obey a direction to move toward a specified destination by moving in the appropriate direction with a mood-appropriate gait. If the puppet feels peppy or the destination is close, it will follow a direct path all the way to the destination. On different occasions, the puppet’s choices of path and gait between the same origin and destination may vary within the ranges allowed by its mood. On the other hand, if the puppet is tired and the destination is far, it may stop before reaching the destination. A determined player can repeatedly select the same destination, directing the puppet to improvise a complete journey to the destination.

Taking a second example, a puppet will obey a direction to play alone by improvising a sequence of mood-appropriate physical and verbal behaviors. If the puppet's mood should change, autonomously or by its player's direction, the quality of its improvisation also will change. For example, the small puppet's sad play might involve quietly wobbling along the upper steps. When directed to quit, she might say, "All done," in a sad tone of voice. Her happy play might involve high hopping on the pedestals, leaping into the chute, and singing cheerful songs. When directed to quit, she might call out "Finishio!"

Given the behaviors and moods currently implemented in the Animated Puppets system, players and puppets can improvise a variety of simple vignettes. More generally, the system illustrates the kind of improvisational story-making supported by the Improv Puppets paradigm: an intimate real-time collaboration among all of the participating players and puppets.

Although the Animated Puppets were designed for children, many adults also find them engaging. An earlier version of the Animated Puppets has appeared "live" at: the ACM Conference on Human-Computer Interaction (CHI) in May, 1995, where 250 people (two at a time) played with them [10]; at the Stanford Arts and Technology Exhibit (SATI) in June, 1995, where about 50 people played with them; and in many private showings at Stanford. Thus, we speculate that adults may enjoy similar sorts of Improv Puppets applications adapted with suitable content.

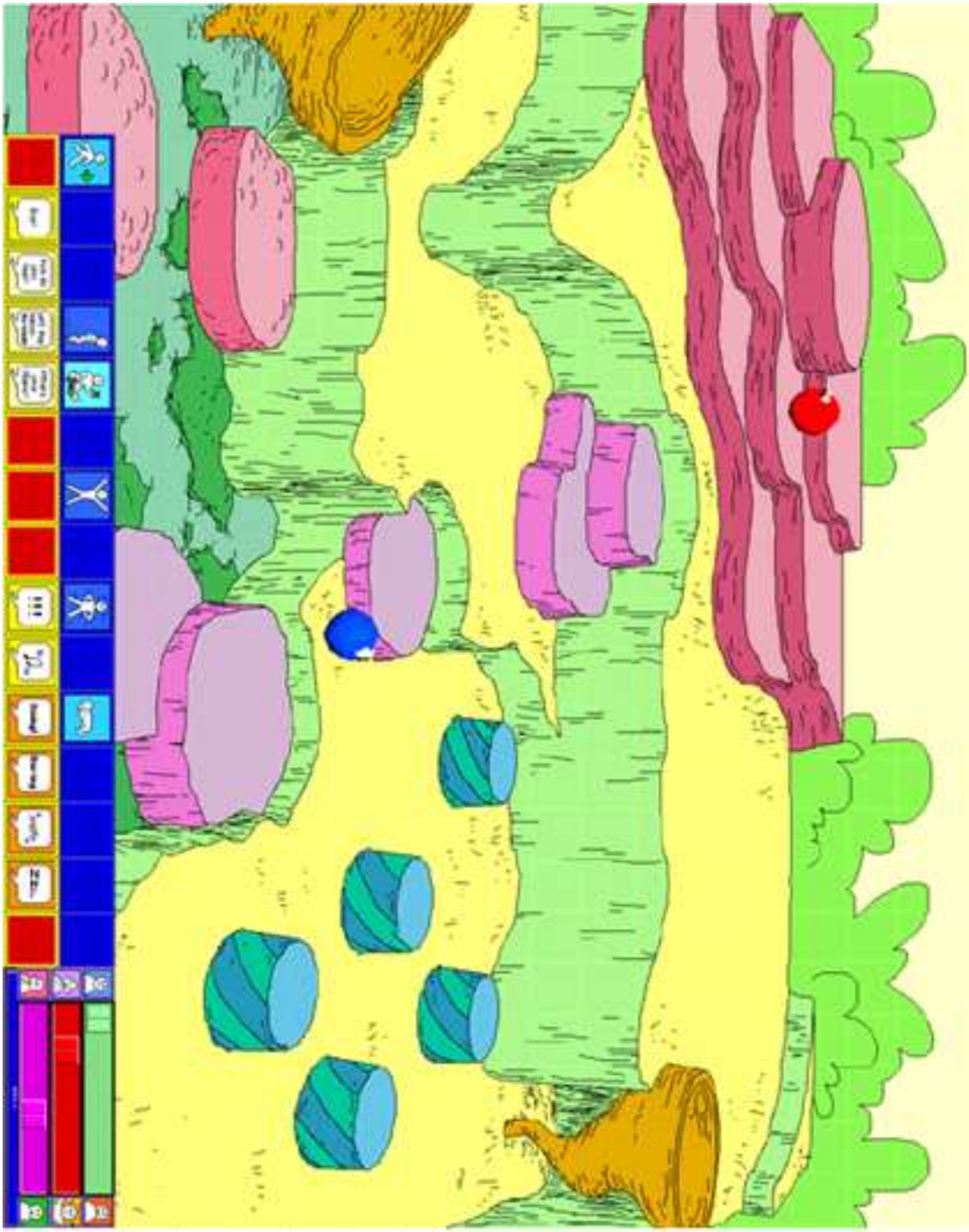

Figure 3. Virtual world and user interface for Animated Puppets.

4. Improv Actors

4.1 Improv Actors Paradigm

"Improv Actors" is a scenario-based interaction paradigm in the style of the commedia dell-arte of Renaissance Italy (subsequently reprised in the Compass and Second City theater groups in the 1950's and 1960's in Chicago). In the commedia, groups of actors entertained audiences by improvising different performances of informal scenarios. A scenario indicated each actor's role, basic function in the plot, and the timing of his or her entrances and exits. Within these bounds, the actor was free to improvise. Similarly, with "Improv Actors," authors create stories by constructing scenarios for the actors to improvise. Based on early improvisations, authors may choose to revise or refine their scenarios for subsequent performances. Improvisational story-making is an iterative scenario-based collaboration between authors and actors. Below we describe a particular Improv Actors application, called "Master and Servant."

4.2 The "Master and Servant" Implementation

The logical and physical system organization for the Master and Servant system is similar to the organization for Animated Puppets, but without the real-time user interface. Each actor's mind (including its mind-body interface) runs as a separate process on a single Sun SPARCStation 10 in our laboratory, while the virtual world runs on a high-end SGI workstation (Indigo II or Reality Engine) elsewhere on the Stanford campus.

For actors' bodies, we use two articulated human figures, Otto and Gregor (the short, squat character and the tall, thin character in Figure 5), which were developed by Ken Perlin at NYU [16]. Both figures can perform physical actions, such as: Walk to (X,Y), Wave, Sit down. The animation system adds life-like randomness to the physical parameters of actions (e.g., the exact position of the arm while waving). Its original user interface (menus for users to command the figures' actions) was replaced by a mind-body interface that allows agents' minds to command their own actions and to perceive one another's actions. The actors mime their interactions in a blank virtual world. A musical accompaniment reinforces the plot and emotional tenor of their performances.

The actors' minds instantiate the architecture described above. Working within the constraints of a scenario provided in advance, they iteratively identify relevant behaviors, choose among them, and execute selected behaviors. During a performance, the actors operate autonomously, without user intervention.

Each actor has a class hierarchy of physical behaviors, each with relevance conditions and mood values (discussed below). Leaf behaviors have execution scripts that can be instantiated, interpreted, and executed. Simple scripts directly command instruction sequences to the animated bodies. Complex scripts (e.g., "Help the master sit down") recursively invoke sequences of instantiated behaviors from the hierarchy. Script parameters permit different improvisations.

The class hierarchy has two main branches defining appropriate behaviors for master and slave roles. Thus, either actor can be directed to play either role.

Each actor has three bipolar, continuous-valued mood dimensions based on the "status" concepts of Keith Johnstone [11]. High status in "demeanor" leads to an erect posture and dignified movements; low status leads to an awkward posture and graceless, unnecessary movements. High status in "relationship" leads to an authoritative stance toward another individual; low status leads to acquiescence. High status in the "space" leads an actor to enter and make use of the available space; low status leads an actor to avoid intruding upon the space.

The status variables are independent and each actor can be directed to adopt arbitrary values on each variable or to change values at particular moments in a performance. Of course, we associate particular sets of values with stereotypic roles, such as master and servant. For example, a master—by definition—has the upper hand in the relationship and owns the space. We expect him to be high status in demeanor as well, but this is not definitional. Conversely, a servant is—again by definition—low status in the relationship and in the space. He might be high or low status in demeanor. As Johnstone observes, however, deliberate manipulation of status variables—sometimes even in contradiction of definiational stereotypes— often leads to interesting and amusing stories.

4.3 Story-Making with Improv Actors

Players collaborate with Improv Actors as they might with human actors, in an iterative story-making process. Players construct abstract scenarios that direct the actors' roles, moods, and actions. The actors collaborate with one another on scenario improvisations during "rehearsal." Based on the actors' improvisations, players might revise the scenario for subsequent rehearsal or performance.

To illustrate, we discuss three variations on a classic master-servant scenario, which we call "When the Master's Away ..." The base scenario has five "blocks:" 1. Master and Servant, 2. Servant at Play, 3. Caught in the Act, 4. Servant Retreats, and 5. Business as Usual. Our user interface allows players to create and refine scenarios such as this one by directing each actor's role, status, and actions in a sequence of scenario blocks. Figures 4a-c show the three scenario variations: Low-Status Servant, High-Status Servant, and Reversal. The implemented agents described above can improvise within all three variations.

Players direct Imrov Actors' roles and status graphically. For example, in Figure 4a, Otto (the short, squat character) is assigned the master role and high status in demeanor, relationship, and space throughout the scenario. Gregor (the tall, thin character) is assigned the servant role and low status in demeanor and relationship. He is assigned low status in the space when the master is present—

by definition, the master owns the space. But he is directed to raise his status in the space upon Otto's exit—allowing him to move out into the space to play—and to maintain high status in the space until Otto confronts him. In keeping with his low-status demeanor, Gregor is directed more specifically to raise his status in the space gradually, so that he hesitates and moves tentatively into the space, and then to let his status plummet when Otto confronts him.

Players direct Improv Actors' actions in a language comprising cues and behavioral directions like the ones given to Improv Puppets. For example, in block 1 of Figure 4a, Otto is directed to perform a specific behavior, "stand at the window," for 5 seconds, then to improvise behaviors within the maximally general class, "do something," for 2 minutes, and then to perform a specific behavior, "exit." During the same block, Gregor is directed to perform a specific behavior, "stand at the wall," for 5 sec, and then to perform any behaviors within the maximally general class, "do something," until Otto exits.

Improv Actors use their directions, perceptions, knowledge, and state information to identify and choose among relevant behaviors in the dynamic situation. For example, after the initial 5 second pause in block 1, both Otto and Gregor will improvise under their general directions to "do something." Otto's role as master and his high status values maximize his improvisational discretion. He can perform whatever masterly behaviors are in his repertoire, including individual behaviors and interactions with his servant. By contrast, Gregor's role as servant and his low status in the relationship and the space limit his improvisational discretion. Even given the maximally general direction to "do something," he must obey his master's orders and otherwise remain in his place near the door. Moreover, given his low demeanor status, Gregor must serve with humility: a slouching posture and awkward movements. On the other hand, in block 2, when Otto is absent, Gregor is released from his low status in the relationship and directed to raise his status in the space. Given exactly the same direction to "do something," Gregor is free to move out into the master's space (albeit hesitantly) and to perform actions ordinarily reserved for the master.

The left column in Figure 5 shows four snapshots from one of Otto's and Gregor's improvisations of the Low-Status Servant variation: a low-status Gregor serving Otto; a low-status Gregor sitting nervously in Otto's chair; a chagrined Gregor confronted by Otto; and a chastised Gregor back in his proper place.

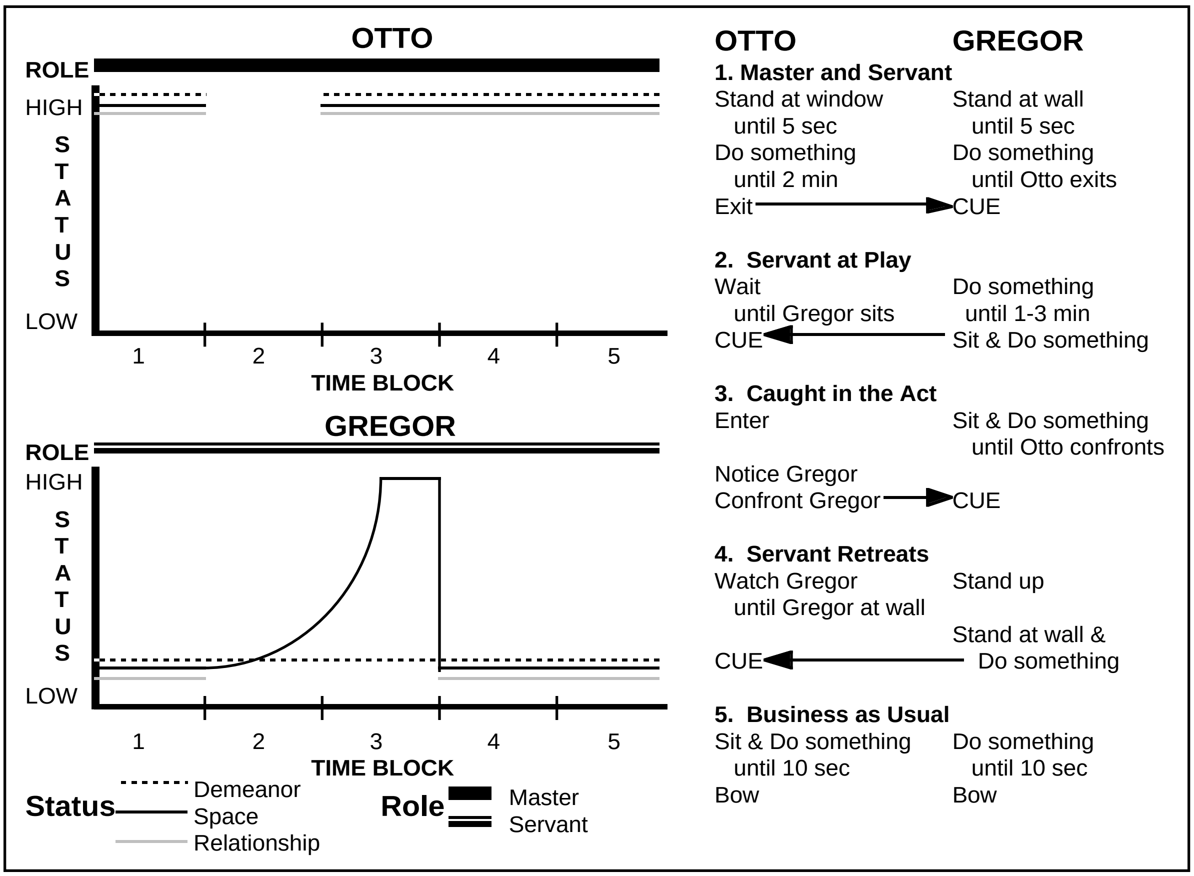

Figure 4a. Scenario for "When the Master's Away ..."
Variation 1: Low-Status Servant.

The High-Status Servant scenario (Figure 4b) differs from the Low-Status Servant scenario only in Gregor's status directions. He is assigned high demeanor status throughout the performance. He still must serve, but now he serves with dignity—erect posture, graceful gestures, and stillness in repose. In keeping with his improved demeanor, Gregor is directed more specifically to raise his status in the space immediately upon Otto's exit and to allow it to decline gracefully following Otto's return.

The middle column in Figure 5 shows four snapshots from one of Otto's and Gregor's improvisations of the High-Status Servant variation: a high-status Gregor serving Otto; a high-status Gregor sitting calmly in Otto's chair; a matter-of-fact Gregor facing Otto; and a nonchalant Gregor back in his place.

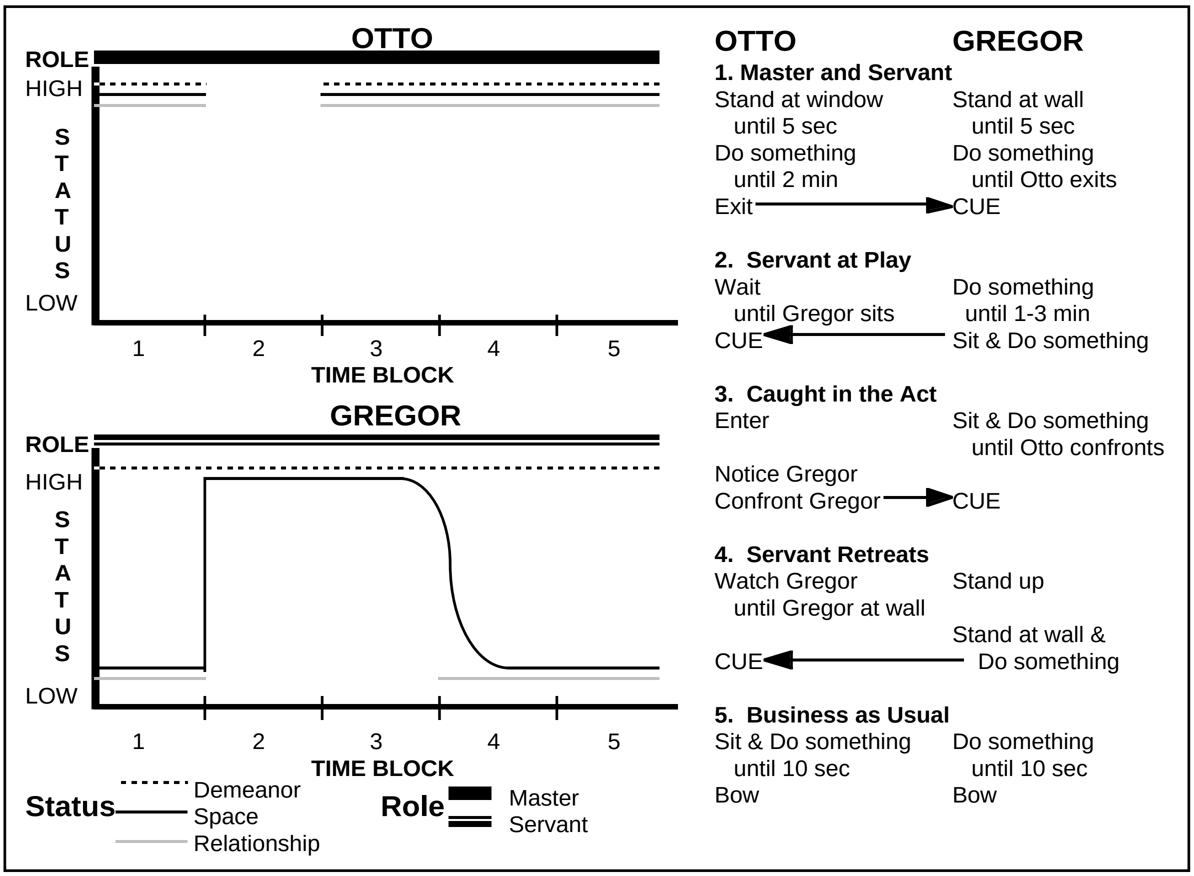

Figure 4b. Scenario for "When the Master's Away ..."
Variation 1: High-Status Servant.

The scenario for the Reversal variation (see Figure 4c) is identical to the Low-Status Servant scenario up to the moment when Otto confronts Gregor in his chair. Instead of cowering under Otto's disapproval, however, Gregor is directed to stand and to sustain the confrontation. Implicitly (in the mind of the director) recognizing his own physical advantage, Gregor is directed to allow his demeanor status to rise, followed by his relationship status, while maintaining his high status in the space. Conversely, Otto is directed to invert Gregor's status dynamics by allowing his own demeanor status to fall, followed by his status in the relationship and the space. These gradual status transitions culminate in an explicitly directed role reversal for both actors (in the graphs) and a specific action direction to Gregor to order Otto to the wall—an overt masterly action.

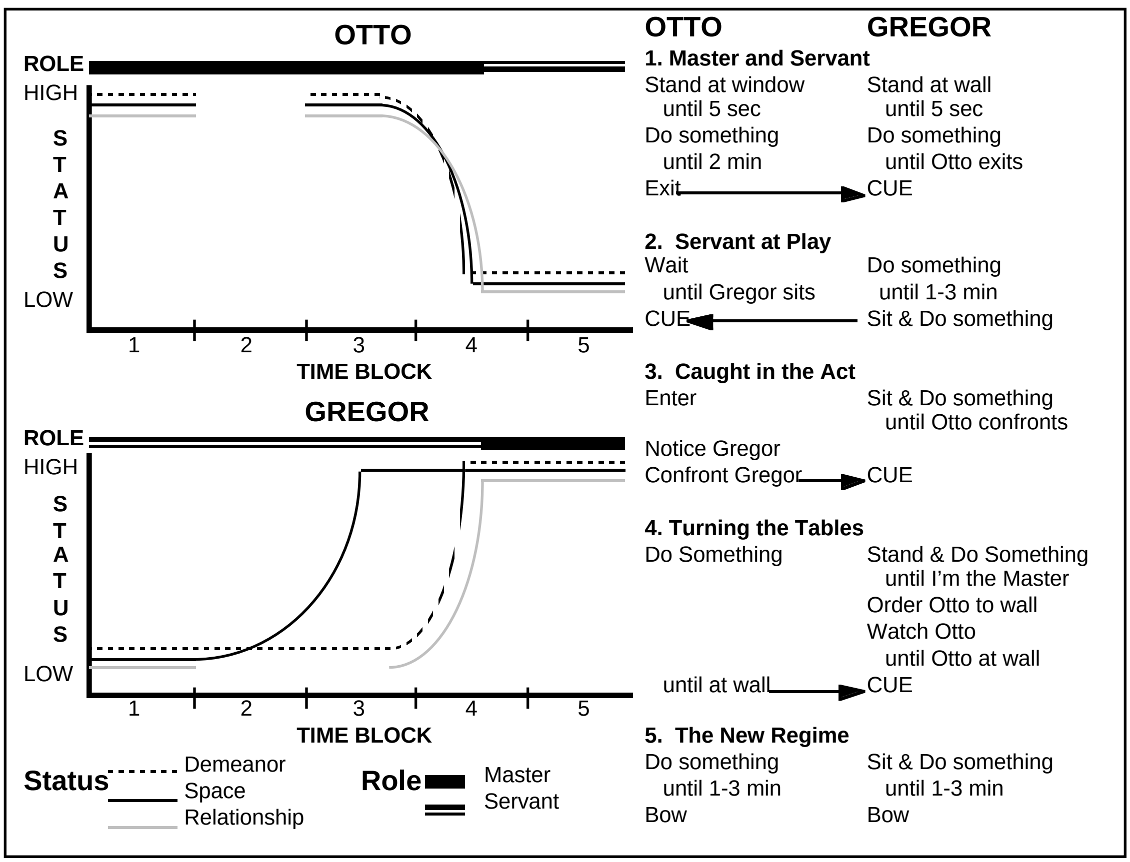

Figure 4c. Scenario for "When the Master's Away ..."
Variation 3: Reversal of Role and Status.

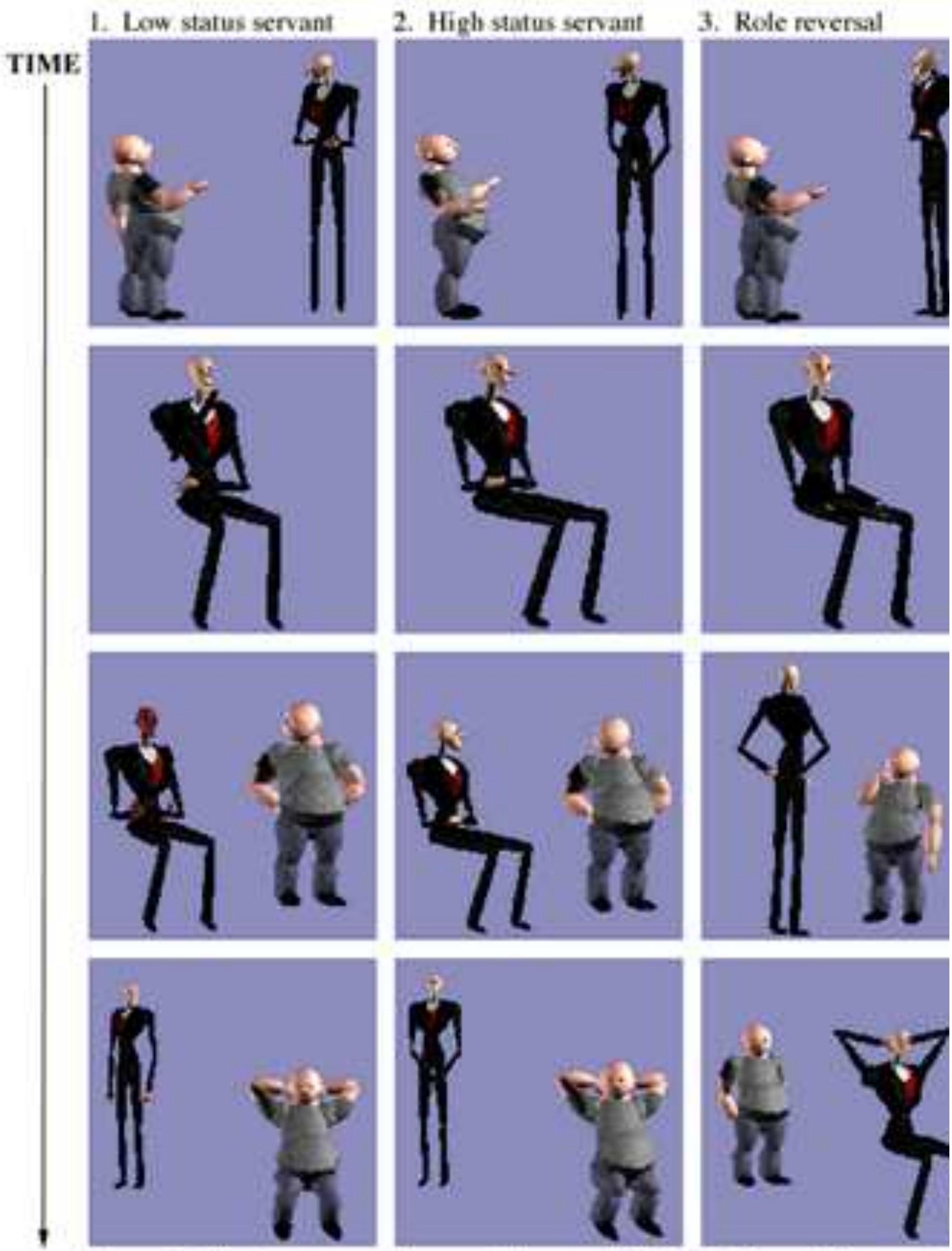

Figure 5. Snapshots from three variations on "When the Master's Away...."

The right column in Figure 5 shows four snapshots from one of Otto's and Gregor's improvisations of the role-reversal scenario: a low-status Gregor serving Otto; a low-status Gregor sitting awkwardly in the master's chair; a defiant Gregor standing up to Otto; and a diminished Otto put in the servant's place.

In sum, improvisational story-making with Improv Actors is an iterative, scenario-mediated collaboration among players and actors. Given the behavioral capabilities, roles, and status variables currently implemented, the Master-Servant agents can collaborate in the creation of a variety of stories in addition to the sorts of variations illustrated here for "When the Master's Away ...."

5. Conclusions

Directed improvisation by computer characters offers several attractive properties as a foundational technology for human-computer interaction. First, it provides a framework in which intelligent computer characters can cooperate with users and one another to achieve goals. Minimally, it offers an efficient communication mode, in which users need not explicitly instruct characters in all of the details incidental to performing a task. Ideally, it offers characters who bring expertise to the task at hand and enhance achievement of the objectives. Second, it explicitly allows flexibility in the manner in which goals can be achieved. Users can give abstract directions, within which computer characters improvise behavior that is compatible with run-time conditions and potentially interesting in other incidental aspects. Third, it provides a natural and familiar style of interaction that mimics the improvisational quality of most human behavior and interaction. In contrast to the stereotypic "feel" of most human-computer interaction, directed improvisation has the variable, idiosyncratic, slightly unpredictable, "give-and-take" of human interaction. Finally, directed improvisation introduces an amusing, engaging, delightful quality to interactive experience in the form of computer characters who combine task-relevant obedience with task-compatible improvisation.

Improv Puppets and Improv Actor represent two different interaction paradigms supported by improvisational agents. In turn, these interaction paradigms support a variety of specific applications, illustrated by the Animated Puppets and Master-Servant systems we have implemented and discussed in this paper. In ongoing research related to these systems, we are using them as prototype environments for children's learning through creative play [1, 3, 12, 13, 14]. In another study, we are investigating how different physical configurations of system components might impact on the nature of the collaborative process and product [8]. Other potential applications of improvisational agents include: multi-player games [6], avatars for on-line environments, character-based computer toys, interactive stories [2], and high-level authoring environments for animated films [15].

References

1. Baker-Sennett, J. Matusov, E., and Rogoff, B. Sociocultural processes of creative planning in children's playcrafting. In P. Light and G. Butterworth (eds.), Context and Cognition: Ways of Learning and Knowing. NY: Harvester Wheatsheaf, 1992.

2. Bates, J., Hayes-Roth, B., and Maes, P. Working notes of the AAAI Symposium on Interactive Story. Menlo Park, CA: AAAI Press, 1995.

3. Flower, L., and Hayes, J.R. The dynamics of composing: Making plans and juggling constraints. In L.W. Gregg and E.R. Steinberg (eds), Cognitive Processes in Writing. Hillsdale, NJ: Erlbaum, 1980.

4. Hayes-Roth, B. Agents on stage: Advancing the state of the art of AI. Proc. of the Int. Joint Conference on Artificial Intelligence, Montreal, 1995.

5. Hayes-Roth, B. An architecture for adaptive intelligent systems. Artificial Intelligence, 329-365, 72, 1995.

6. Hayes-Roth, B. Directed improvisation: A new paradigm for computer games. Proc. of the Computer Game Developers' Conference, Santa Clara, CA: 1995.

7. Hayes-Roth, B. Opportunistic control of action. IEEE Transactions on Systems, Man, and Cybernetics, 12, 1575-1587 1993.

8. Hayes-Roth, B., Brownston, L, and van Gent. Multi-agent collaboration in directed improvisation. Proc. of the First Int. Conference on Multi-Agent Systems. San Francisco, 1995.

9. Hayes-Roth, B., Pfleger, K., Lalanda, P., Morignot, P., and Balabanovic, M. A domain-specific software architecture for adaptive intelligent systems. IEEE Transactions on Software Engineering, in press, 1994.

10. Hayes-Roth, B., Sincoff, E., Brownston, L., Huard, R., and Lent, B. Directed improvisation with animated puppets. Proc. of CHI '95 Conf. on Human-Computer Interaction. Denver, 1995.

11. Johnstone, K. IMPRO: Improvisation in the Theatre. NY: Penguin Books, 1987.

12. Malone, T.W., and Lepper, M.R. Making learning fun: A taxonomy of intrinsic motivations for learning. In R. E. Snow and M. J. Farr (eds), Aptitude, learning and instruction. Hilssdale, N.J.: Erlbaum, 1987.

13. McCaslin, N. Creative Drama in the Intermediate Grades. Studio City, CA: Players Press, Inc. 1987.

14. Papert, S. Mindstorms: Children, computers, and powerful ideas. New York: Basic Books. 1980.

15. Parisi, P. The New Hollywood: Silicon Stars. Wired, December, 1995.

16. Perlin, K. Real-time responsive animation with personality. IEEE Transactions on Visualization and Computer Graphics, 1, 1995.

â€¢ Hayes-Roth, B. and van Gent, R. Story-making with improvisational puppets and actors. Stanford Knowledge Systems Laboratory Report KSL-96-05, 1996. â€¢ Hayes-Roth, B., Brownston, L., Sincoff, E., and van Gent, R. Directed improvisation by computer characters.Â â€¢ Motion tracking made it possible for participants to turn themselves into virtual bats, flapping their arms to fly around a castle inhabited by interacting Improv characters. â€¢ The more severe Gregor character from this scene joined Otto, and the Improv system of this period, to make possible Barbara Hayes-Roth's "Master/Servant Scenarios" at Stanford University. Story-Making with Improvisational Puppets - "Improv Puppets" is an interaction paradigm modeled after traditional puppet play. With physical puppets, children move their puppets' bodies and speak their puppets' words in real time. Thus, the puppets serve as passive vehicles for the children's collaboration on improvisational story making. Similarly, with Improv Puppets, players direct the physical and verbal behavior of their puppets in real time. Now, however, the puppets are synthetic agents. They are animated and smart.Â Thus, improvisational story making becomes a collaboration among children and puppets. This paper describes the design and behavior of an implemented system that allows two children to direct the improvisations of two puppets in real time. Improvisational theatre, often called improvisation or improv, is the form of theatre, often comedy, in which most or all of what is performed is unplanned or unscripted: created spontaneously by the performers. In its purest form, the dialogue, action, story, and characters are created collaboratively by the players as the improvisation unfolds in present time, without use of an already prepared, written script.