Computer-Aided Exploration of Virtual Environments

Patrick Doyle and Barbara Hayes-Roth

Knowledge Systems Laboratory
Stanford University
Gates Computer Science Building 2A
Stanford, CA 94305-9020
pdoyle@cs.stanford.edu, bhr@hpp.stanford.edu

Abstract

We introduce an intelligent "guide" that assists users in learning about and exploring in virtual environments, particularly MUDs. The guide itself acts independently of the user, while the user controls his or her own avatar. In addition to accepting options offered by the guide, our system allows the user to direct the avatar by explicit MUD commands as well as via abstract instructions and constraints. The guide, in addition to scanning the MUD environment for actions and knowledge to present to the user, can communicate with the user, suggest activities, and record user preferences to tailor its behavior across multiple sessions.

1. Introduction

The Wart did not know what Merlyn was talking about, but he liked him to talk. He did not like the grown-ups who talked down to him, but the ones who went on talking in their usual way, leaving him to leap along in their wake, jumping at meanings, guessing, clutching at known words, and chuckling at complicated jokes as they suddenly dawned. He had the glee of the porpoise then, pouring and leaping through strange seas.

"Shall we go out?" asked Merlyn. "I think it is about time we began lessons."

-- T. H. White, *The Once and Future King*

As computers have become commonplace in homes and educational institutions, designers have searched for ways they can assist in the learning process. The rise of Internet connectivity is beginning to provide a range of interactive, educational environments for users to explore and learn from. Our goal is to provide an interactive, intelligent agent as a guide to help the user understand and navigate these environments.

Our specific interest is with children exploring virtual environments both for learning and entertainment. Existing educational text-based environments include "virtual classrooms," which simulate real classrooms in an electronic setting, as well as unreal environments that have strong entertainment or exploratory appeal, such as models of the human body users can walk through, online museums or galleries, and games.

As part of our Virtual Theater project, we are creating a guide to explore these places with a child. The guide itself is "intelligent" and possesses its own personality. It can offer comments, suggestions about things to do or see, give the child special abilities to interact with the environment, and explain in greater detail what the child is seeing and experiencing.

We believe our approach has several appealing aspects. It is an excellent testbed for research on intelligent agents, interactive agents, child development, and virtual environments. It provides an enjoyable way to explore experience-based learning environment, supporting both the educational goals these particular environments and the development of social skills. Lastly, learning with a computer-controlled guide is a natural vehicle for demonstrating and examining social interaction as an HCI paradigm.

The rest of this paper is organized as follows. Section 2 introduces our guide agent and its goals. Section 3 explains the kinds of environments we want the agent to operate in. Sections 4-5 describe the abilities of the guide agent and the user's avatar. Section 6 explains the user interface of the system. Section 7 describes how the underlying architecture operates, and our future plans for development. Section 8 briefly outlines related work, and Section 9 discusses social interaction between the user and the guide as a paradigm for HCI.

2. Introducing the Guide: Goals and Motivations

Our guide is called Merlyn, after T. H. White's befuddled seer in *The Once and Future King* (White, 1958). In White's retelling of the Arthurian legends, Merlyn guides a young Arthur (whom he calls "Wart") through a series of experiences designed to teach him about the world and train him to become a king. Through the help of Merlyn's magic, Wart becomes a fish, an ant, meets King Pellinore and the legendary Questing Beast, and other wonders. The virtual environments of today can offer similarly varied, intriguing, and educational experiences to real children.

Merlyn is intended to be a playmate, a teacher, and a friend to the exploring child. His purpose is to make the MUD environment easier to understand and more engaging to explore. There are four main tasks that Merlyn performs to accomplish this.

First, Merlyn is an anthropomorphic friend to the child. He is presented as a kindly old man, occasionally confused[1], sometimes wise, and invariably faithful. Bates and others have written on the importance of emotion for creating "believable agents" (Bates, 1994). Merlyn has distinct, constant personality traits, such as kindliness, governing his behavior, and in addition he has transient moods (such as happiness or depression) that influence his actions. The combination of these create an emotionally-complex friend for the child to interact with.

Also, the virtual environments we are exploring lend themselves exceedingly well to anthropomorphizing Merlyn. Since he appears to the child just as everyone else does – as an animated graphic, or as a string of text messages on a screen – the illusion of another person is a strong one. Julia, the infamous MUD robot, carried on conversations with other MUD users, and even when they found out she was a computer program some users felt that she was a friend (Mauldin, 1994).

Merlyn's second task is to inform the child about the environment. He[2] has sensors that scan the environment to learn what actions are currently available and what details are knowable. Ideally, he will request data from the MUD which will provide them in a format he can then make available to the child on demand. We foresee creating *annotated environments* that provide such data meant specifically for exploratory agents, to enable those agents to organize and present information that would not naturally fit into standard descriptions.

[1] This personality trait is a distinct advantage in an AI agent.

[2] We have already fallen into our anthropomorphic trap.

As an example, suppose Merlyn and Wart are visiting an historical museum. They are standing in a room exhibiting King John's signing of the Magna Carta. Merlyn, scanning the room, is told that "King John" and "Magna Carta" are important educational elements, and he requests more data from the room on each. He then can present to the child the option "Tell me more about King John" as an action. If the child should choose the action, Merlyn will relate what he has learned – perhaps biographical details on John's life, his ambitions for England, the prevalent social conditions leading to Magna Carta, and other related topics. The nature and complexity of the information would be up to the environment's creator, but it can be presented through Merlyn's friendly and decidedly idiosyncratic personality.

Merlyn's third task is to suggest activity to the child. By keeping a record of places he has seen and actions he knows are available, he can offer to go back to these places again. Either when requested by the child or as part of his improvisational ability, Merlyn can make suggestions about fun or educational things to do. ("Wart, I believe we should visit the Barber of London to learn more about medieval medicine.").

Lastly, Merlyn should be able to play with the child. "Play" is a loosely-defined term ranging here from simple conversation to engaging in multi-player activities in these virtual environments, such as a game of chess. Merlyn's ability to perform complex tasks will be strictly limited by the information he can take from the environment, but we hope to give him enough instructions to make these behaviors interesting.

3. Merlyn's Habitat: Virtual Environments

As homes, educational institutions, and other organizations have increasingly become connected to the Internet, interest has arisen in network-based, multi-user virtual environments. Such systems provide evolving and dynamic environments in which users, whether children or adults, can explore and learn both by conversing with others and through the use of interactive games and tools contained in these environments.

One kind of such environments are MUDs (Multi-User Dungeons or Multi-User Dimensions). These text-based systems allow many users simultaneously to connect to virtual "worlds" composed of rooms, objects, and people. Depending on the design of a particular system, themes vary from fantasy environments with dragons and wizards, to futuristic exploration with spaceships and aliens, to university settings complete with whiteboards, lecturers, and regular conferences. MUDs provide generic, programmable environments to support a wide range of interactive applications.

```
> look
On the Stage
   This is the large wooden stage of a small theater used for improvisational
experiments.  The stage is bare but well-lit by spotlights high above.  The
plush theater seats rise up all around the stage into the darkness.

-- There is one obvious exit:  offstage.
William Shakespeare, the Bard of Avon.
A banana cream pie.
> examine spotlights
High above the stage, brilliant white spotlights illuminate the actors.
> get pie
Ok.
> eat pie
It doesn't look very appetizing.
> say Hi, Will!
You exclaim: Hi, Will!
William Shakespeare says: A most amiable day to you, sir.
>smile shakespeare
You smile happily at William Shakespeare.
William Shakespeare bows with a flourish.
```

Figure 1. A Typical Interaction on a MUD

Within a MUD, each user may pick up and manipulate objects, move from one location to another, interact with other beings, both alive and computer-controlled, or perform more complex actions such as playing games, sending electronic mail, or creating new objects. Figure 1 gives a typical example of interaction in a MUD.

Our current environments for exploration are MUDs. As they are text-based, they have the advantages of being easy for the agent to parse and communicate with, and they also relieve the designers (that is, us) from creating complex graphics and animations in order to communicate in the medium. Graphical variants on such worlds are beginning to become popular as well, and in time we see no reason why Merlyn's architecture cannot be transferred to new, more visually exciting interfaces.

The child will act on the MUD through Wart. Unlike Merlyn, however, Wart only speaks or acts with the child's guidance. Wart merely records what he sees, offers the child high-level commands via the agent architecture, and helps the child communicate with Merlyn. From the child's point of view, Wart is not an independent entity; instead, he is an invisible character that provides a channel for high-level communication between the child and the environment.

The advantage of using our Virtual Theater systems to control Merlyn is that they bring with them capabilities for interaction and improvisation. Merlyn is capable of choosing actions either by being given high-level requests by the user or on his own initiative. He has moods and interests of his own, as well as learning the likes and dislikes of the user he is working with, and uses these to guide his decision-making and improvisational choices. The result is a more lifelike, friendly, and entertaining way of viewing the virtual world.

4. Merlyn's Abilities

Merlyn is operated on the virtual environment across an ordinary socket connection by the Virtual Theater system. To the MUD, he is just another character driven from a remote location. However, his Virtual Theater "brain" gives him a variety of abilities to explore, to learn, and to adapt to his environment and his user.

The basic traits Merlyn takes from the Virtual Theater system are his personality, moods, and ability to evaluate and decide upon appropriate actions in context. Context is created by the personality, mood, user requests, and environment. However, unlike ordinary Virtual Theater agents, Merlyn has several additional features designed to make him flexible in an unknown environment, and specifically to help him interact with an individual user.

First, Merlyn can scan the environment. By "scanning," we mean that he can request from the environment a list of valid actions and a list of things that can be examined. If the MUD provides additional hooks to provide ontological knowledge of the environment to an agent, our intention is for him to use them as well. (Indeed, in our experiments we plan on incorporating such knowledge in a testing environment.)

Merlyn incorporates these actions into an action tree structure. Whenever a new action is discovered, it is added to the structure. He associates with new actions their location and, when possible, their meanings. Data about features of the environment (for example, the description of the Forest Sauvage, factoids about ant military structure, and the like) are stored in a keyword database for later retrieval and presentation to the child.

In addition to storing information about the environment, Merlyn also records Wart's actions. In this way he has an expanding database about what the child has done, and can extrapolate to suggest repeating old activities or trying new ones he has learned about that have similar characteristics. The child can also explicitly tell Merlyn that an activity on the MUD is interesting or uninteresting, and Merlyn will rank-order these actions accordingly when he offers things to do.

Also, through the process of offering activities, Merlyn receives feedback from the child. This feedback will be used to weight Merlyn's decisions. So, if Merlyn offers to take the child to a museum, and the child indicates that museums aren't interesting, he will turn to other activities in the future. He can also be preset with weights and suggestions for a particular MUD if Merlyn is being used to achieve particular educational goals.

5. Wart's Abilities

As we described earlier, the child interacts with the MUD through Wart. Wart is a conceptual, rather than a physical, entity; to all intents and purposes to the child there is no agent present. As a result Wart's features are limited to gathering user data and presenting choices.

From the child's point of view, Wart is only a menu of options provided with the interface. Wart's job is to control the options that appear, which change from area to area and from time to time. The child can either type directly on the text-based interface MUD, bypassing Wart, or use the list of options to invoke high-level commands through Wart.

Wart does have moods, however. His moods are meant to reflect the child's moods, and the child has control over them. Depending on Wart's mood, offered actions may change from "Greet Merlyn" to "Play a Trick on Merlyn" or even "Kick Merlyn." The configurable moods allow the child some control over what appears as suggestions and options. Multiple, interacting moods give several axes on which to prioritize actions, so they can apply not only to social interaction but to presenting actions related to playing games, exploring, learning, and the like.

Beyond this, Merlyn has control over Wart. He can direct Wart to provide new or different choices to the child. In this way the interface is both Wart's and Merlyn's gateway to the child. They also mean that Merlyn can *endow* (to borrow a term from improvisational theater) Wart with abilities he does not ordinarily possess. Our expectation is that while Wart will have a fairly small, consistent repertoire of actions to provide, Merlyn will drive Wart to offer choices relating to information retrieval (e.g., "Tell me about owls"), multi-player activities ("Let's become fish") and other options Merlyn gathers from the environment itself.

6. The Interface

The interface for our system has two goals. First, it provides a window directly onto the virtual environment, currently a text-based MUD. Second, it provides support for Wart's suggested commands, Merlyn's suggestions, and any other information Merlyn wants to present beyond the environment itself. In the current system we have decided on a simple interface, handling most of Merlyn's discourse via the MUD environment.

Although the interface between the user, Wart, Merlyn, and the MUD could be implemented as a simple text window alone, we have chosen to make a specialized interface. The text window does not offer enough control to make the interface easy to read and therefore does not fit well with our target audience.

The windowed interface consists of two parts, the text window and the menu buttons. The text window is an ordinary connection to the MUD, displaying whatever the MUD displays and allowing the user to type commands directly to the MUD in it. In this way the child does not feel the agents are hampering him or her from exploring, and the sense of direct engagement is maintained.

The menu buttons are where the agent intelligences are presented. Button options are a suggested list of current actions for the child; pressing the button invokes the action. Actions can either be atomic, and therefore mapped directly onto the MUD (such as "Smile") or can involve more complex lists of commands in the MUD environment (such as "Go to the Forest").

Menu button options change from one environment to another and also from time to time. Options are under the direct control of the Wart agent. He lists whichever actions he believes are most appropriate for the child in the current context. When the context changes – by moving to a new location, encountering a person, picking up or putting down an object, or by the child changing Wart's moods – then the list of options is updated.

Merlyn also has control of the option list, though indirectly. He can direct Wart to suggest activities to the child explicitly, or he can simply make some of Wart's known but unavailable actions available, in which case

Wart can decide whether they should be listed. This is how Merlyn can endow Wart with abilities temporarily; portions of Wart's permanent list of actions may be available only when Merlyn turns them on, and become unavailable when Merlyn turns them off. Explicit direction is used in cases where it is not Wart's action, but Merlyn's, that is relevant.

7. Implementation and Future Plans

The guide system is presently in the design stage. Critical to our success are the issues of representation of annotation, flexibility of the action trees used by Merlyn and Wart, and the structure of Merlyn's database for easy access and coordination of the information he retrieves. Here we briefly outline the design of the system.

To support the guide system, the Virtual Theater architecture performs three major operations. It operates two intelligent agents, one for Merlyn and one for Wart. It maintains simultaneous network connections to the MUD environment for each agent. Lastly, it operates the visible interface presented to the user.

The overarching system architecture coordinates communication between the agents and the MUD environment via network connections, so that the agents are informed of all incoming data and they can issue commands directly to the MUD. It also coordinates agent commands to the interface, which control the appearance of the actions menu, and the user's replies in the interface. If the child types in the text window of the interface, that text is sent directly to the MUD environment through the Wart avatar; otherwise, if the child chooses one of the action buttons on the options menu, that request is passed to the agents for processing.

Agents in the system use the BB1 dynamic control system as their cognitive architecture (Hayes-Roth, 1985). BB1 essentially runs a three-step cycle. First, an agenda manager notices behaviors appropriate to the situation triggered by current perceptions and reasoning. Second, a scheduler chooses the best available behavior based on the control plan, introducing some controllable "noise" in evaluating the most appropriate behavior. Lastly, the executor executes the chosen behavior.

The control plan is a key feature of BB1. Since it can be as abstract as desired, the scheduler can choose among several acceptable behaviors. There can be multiple simultaneous control plans, whose combined weights determine which action is best. Each individual plan can have its own structure and may be weighted by a variety of factors, including user input, current mood, personality, and environment. Since the control plan itself is a data structure, executing actions can modify it, thereby changing criteria used to schedule future actions.

Earlier Virtual Theater project architectures have successfully been used in animated, graphical improvisation and "puppet" systems (Hayes-Roth, et al., 1994). The new features added in Merlyn and Wart revolve around scanning environments, modifying action choices to accommodate new actions, and storing acquired knowledge in databases for later access.

In this work, we intend to collaborate with educational MUD designers to include annotation in their environments that Merlyn can use to advise the child. We also intend to generate MUD environments of our own for Merlyn and the child to explore. The more annotation, the greater Merlyn's flexibility and understanding of what he sees. Initially, annotations in MUD environments would be keywords and action lists that Merlyn could absorb directly into his database and action choices. Eventually, we want to enable Merlyn to obtain instructions for processes from the environment so that, for example, he could learn the rules to encapsulated games by asking the system to supply them.

8. Related Research

For some time research has been done on the social aspects of virtual communities and on MUDs in particular (Curtis, 1992; Curtis and Nichols, 1994). In addition to studies of how human beings interact in text-based, virtual environments, research has been done specifically on the issue of robots in MUDs. Mauldin created Julia, an agent that freely wandered on a TinyMUD, exploring the environment, having limited conversations with other users, providing directions, and occasionally fooling users into believing she was a real human (Mauldin, 1994).

Julia was tailored specifically for existing in a MUD environment. Other projects, most notably Joseph Bates' Oz project at Carnegie-Mellon University and Pattie Maes' autonomous agents at MIT have focused on creating believable (Bates, 1994) and lifelike (Maes, 1995) intelligent agents, both for entertainment and general purposes.

We are attempting to unify these lines of research, by combining the naturally computer-based interaction of virtual communities with a believable, lifelike agent. Our Virtual Theater architectures allow us to create intelligent, behavior-driven characters that have the autonomy and complexity to behave in useful and entertaining ways. We can put these characters in virtual communities which are themselves expanding and developing, and allow them to interact with other people and new situations, so that they are not locked into pre-scripted scenarios, but can advance with their environments.

As a side issue of interacting in unknown environments and providing useful data about those environments to the user, data the agent could not create itself, we are interested in allowing that agent to acquire knowledge from its environment via annotation, and present what it learns in an easily-digestible format to the user. In this respect our work is similar to the Guides project at Apple that sought to use interactive, though non-intelligent, guides to present information from Grolier's Encyclopedia that related to early American history (Oren, et al., 1995).

9. Social Interaction as an HCI Paradigm

Some recent research in human-computer interaction (Nass, Steuer, & Tauber, 1994; Nass, et al., 1995) has suggested that human interaction with computers is fundamentally social. Nass describes this as the "computers are social actors" paradigm (Nass, et al., 1995). That is, the social rules we use in dealing with humans are applied in our interactions with computers as well.

Our guide system of Merlyn and Wart provides an excellent framework for exploring this paradigm. The child's interaction with the computer is explicitly social. The computer, in the form of Merlyn with his querulous comments, or in the form of Wart and his menu buttons, provides conversation and choices relying upon ordinary rules of social interaction for them to be sensible.

Social interaction is naturally more comfortable for users than typing arcane commands at a prompt. This is especially true of our target users, who are children By providing a social atmosphere as a means of interaction, we can increase their sense of direct engagement with the world. We can also strengthen the tendency to anthropomorphize Merlyn, making him into a real character with far more detail than we could ever explicitly provide.

The social atmosphere we provide comes in the form of Merlyn's spoken feedback, and also the nature of the text we provide on the options presented to the child. If the child feels that the option buttons are simply iconic representations of social communications, then the sense of conversation with Merlyn will be natural and direct. Since the goal of the project is to provide the child with a friendly, familiar, and believable guide, we are hopeful that our interface provides this experience.

Acknowledgements

Our work has been 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 Office of Technology Licensing. We thank Lee Brownston, Daniel Huber, Robert van Gent and Daniel Rousseau for suggestions on our design.

References

Bates, J. The role of emotion in believable agents. Communications of the ACM, 1994, 37, 122-125.

Curtis, P. Mudding: Social Phenomena in Text-Based Virtual Realities. In *Proceedings of the 1992 Conference on Directions and Implications of Advanced Computing*. Berkeley, 1992.

Curtis, P. and Nichols, D. MUDs grow up: social virtual reality in the real world. In *Digest of Papers, Spring COMPCON 94*. IEEE Computer Society Press, Los Alamitos, 1994, 193-200.

Hayes-Roth, B. A blackboard architecture for control. Artificial Intelligence, 1985, 26, 251-321.

Hayes-Roth, B., Sincoff, E., Brownston, L., Huard, R., and Lent, B. Directed improvisation. Stanford Knowledge Systems Laboratory Report KSL-94-61, 1994.

Maes, P. Artificial life meets entertainment: lifelike autonomous agents. Communications of the ACM, 1995, 38, 108-114.

Mauldin, M. ChatterBots, TinyMUDs, and the Turing test: Entering the Loebner Prize Competition. In *Proceedings of the AAAI 1994 Conference*. MIT Press, Cambridge, Mass., 16-21.

Nass, C., Steuer, J., and Tauber, E. Computers are social actors. In *Proceedings of the CHI Conference*. Boston, Mass., 1994.

Nass, C., Moon, Y., Fogg, B., Reeves, B., and Dryer, D. Can computer personalities be human personalities? International Journal of Human-Computer Studies, 1995, 43, 223-239.

Oren, T., Salomon, G., Kreitman, K., and Don, A. Guides: Characterizing the interface. In *The Art of Human-Computer Interface Design*, Brenda Laurel, ed. Addison-Wesley, Reading, Mass., 1995.

White, T. H. *The Once and Future King*. Collins, London, 1958.