Proceedings of the 2002 AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment, pp. 103-108

Making the Human Care: On Building Engaging Bots

Robert Zubek and Aaron Khoo

Northwestern University Computer Science
1890 Maple Avenue
Evanston, IL 60201
{rob, khoo}@cs.northwestern.edu

Abstract

Computer games research has focused strongly on improving the tactical intelligence of computer opponents. However, the social aspect of gaming has been largely ignored. We claim that, in many cases of multi-player gaming, the simple knowledge that one's opponents are emotional creatures is far more important than the opponents' skill or intelligence. We also claim that conveying this emotion in online games can be achieved through simple and cheap mechanisms. As proof of concept, we present a non-player character system for the game Counter-Strike that begins work in this direction.

Introduction

Much of the recent research work in AI for action games, such as Half-Life, Quake, or Unreal, has centered on making the bots more realistic and humanlike. In order to increase realism, researchers are investigating systems that approximate naturalistic navigation methods, human perceptual and cognitive limitations, and more recently, systems that simulate the decision processes of a skilled human player. For example, the advanced QuakeBot agent (Laird and van Lent 99) incorporates prediction and learning (Laird 00) in an attempt to approximate the way a human player might try to figure out what the enemy is going to do next. The QuakeBot simulates an expert player's decision processes in the effort to enhance the bot's tactical intelligence, making the bot more realistic and, one hopes, more fun.

However, merely creating a technically skillful opponent ultimately ignores the social aspect of gaming. A survey of any online gaming community reveals that human opponents are always preferred over computer-controlled non-player characters, or NPCs, regardless of how technically sophisticated the computer might be. Indeed, there is a large variation in the skills of opponents one encounters in the online world. However, the experience of playing with other humans is worth the aggravation of unknown opponents, who might be greatly above or below the player's own skill level. Players are willing to tolerate long, frustrating losing streaks, and keep returning for more punishment. Similarly, many players are also happy to contend with novices, or 'newbies', who are often both clueless and hapless. This is interesting because it suggests that the actual intelligence of the opponent as manifested in his game skill might actually be less important than the shared experience of playing the game together with other humans.

Copyright © 2000, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

The importance of being social

We believe the primary attraction of human opponents lies not in their skill or intelligence, but rather in the knowledge that they are real, emotional creatures. When humans lose a game, they feel frustration and pain; when they win, they brag and boast loudly. That is the essence of fun in defeating a living person – imagining your enemy groaning in agony and banging his fists on the table, seeing their annoyed reply in the chat window, being able to rub it in and make their loss that much more agonizing. Conversely, losing against a human is excruciatingly frustrating precisely because you know your opponent is laughing with glee as she beats you down, which fuels an ever growing desire to exact sweet revenge. The emotional involvement is not merely a side effect of the players being human. It is a crucial part of the fun in online multi-player games. It is the key to enjoying the game on a whole new level.

Although this enjoyment may seem visceral, none of this should be surprising. We have to remember that these online games are mostly designed for, and played by, teams of adolescent males. The mode of play thus expectably reflects adolescent male culture, along with all of the natural aggression, competition, and penchant for conflict that fuels it. This is not abnormal – boys will be boys, as they say, and their natural mode of gameplay transfers into online worlds as well (Jenkins 98). Online games are popular with adolescents exactly because they present an environment in which these behaviors are acceptable and reciprocated.

Mature gamers may be disdainful or disinterested in such juvenile behavior; many of us no longer derive pleasure from such social interactions. However, adolescents do not usually play online games for the intellectual or technical challenge alone. They ultimately seek a social world that they enjoy and understand, which inevitably includes ultra-competitiveness, forming clans, trash-talking, swearing at the winners and lording over the losers. It's impossible to ignore manifestations of these behaviors when playing online.

We are not here to debate the suitability of such behavior. That is an argument better suited to others with more qualifications in such areas. Rather, we simply observe the presence of an obvious attribute of online environments, and seek to take advantage of its existence in the development of enjoyable computer characters.

Insofar as a significant part of the enjoyment of an online game derives from its social effects, playing against a computer opponent completely pales in comparison. An NPC, or bot in an action game, feels no humiliation in losing, and no joy in winning. This may be of technical advantage, since a computer never gets sweaty palms during a tight game, but it makes beating a bot a thankless task. In the end, the human player knows that the computer simply does not care. It is hard to derive a sense of satisfaction from the defeat of a soulless entity that plays on a technical level only, and has no emotional investment in its own success or failure.

Emotionally invested NPCs

We would like to suggest that in order to make bots not merely more challenging, but more enjoyable to play, they need to be able to better participate in the social interactions in the game. As a starting point, we can consider simulating the two common and arguably simplest characteristics of players' interactions: emotional involvement, and trash-talking.

Emotional involvement is deceptively simple: the idea is to create a bot that behaves more like a human, not in the sense of *playing* like one but in the sense of *emoting* like one. Players not only feel the frustration of losing and joys of winning, but are quite vocal about expressing them – and that is also what the bot needs to do. In essence, we want the human to feel the emotional involvement of her opponent. The human should think that her opponent is pounding the table in frustration and screaming when he loses. Conversely, the opponent should be happy and boastful when a success occurs during game play. This empathy will tap into the natural competitive aggressiveness of the human player, and help maintain the illusion that the opponent is more than a set of silicon circuits and unfeeling connectors.

This emotional involvement manifests itself through a behavior commonly referred to as trash-talking – essentially exaggerated verbal posturing. Online games are particularly conducive for this mode of communication. In the course of normal daily life, we discern the emotions of others through the use of body language, tone, facial expressions, and a large number of other subtle cues. But in online games, all we have to go on is the exchange of pure text messages, or chatting. Voice communication online is still relatively rare, and most online games still do not allow players' avatars to display facial expressions or body language. Therefore, the only way to communicate in a social interaction is through a series of text messages scrolling past on the screen. Our bot must be able to participate in this exchange, responding to game events and other chat messages with the appropriate emotional import that the human players can detect.

Taking advantage of the domain

Solving such a problem in the general case would call for us to tackle the natural language problem. We would argue, however, that this would be complete overkill. The gaming domain limits the scope of the problem, making it possible to use cheap mechanisms to implement a fast system that works well most of the time, without having to descend into the tar pit of full-blown natural language comprehension.

First, we observe that the human should already be predisposed to expect other humans. When playing online games, players usually expect to be playing with other people, unless they see some evidence to the contrary. For example an opponent that's technically competent, but incommunicative and socially dumb would be a quick giveaway. This inherent expectation of encountering other humans means that we do not need to create a belief of the bot's humanity from scratch. We only have to be competent enough to support it. If the player is predisposed to expect humans, and the bots act in adequately human ways, players should treat them like they treat other players.

Second, doing this imperfectly but adequately should not require expensive techniques. Players are already predisposed to interpret communication in the game as honest attempts at social interaction. Furthermore, these online conversations are neither deep nor well-structured. Chatting in online games often resembles Internet Relay Chat, or IRC, in that the conversations are often:

- Disconnected – topic changes are quite frequent; it is easy to lose track of who is talking to whom and about what.
- Layered – it is common for one person to participate in more than one thread of conversation

simultaneously; conversely, it is easy to miss entire sections of one thread while replying to another.

- Filled with bad spelling, worse grammar, and just awful language.
- Stocked with stereotypical personality types, such as the boaster, the sore loser, or the "are-you-a-chick" lecher.

If online conversations are largely disconnected, multithreaded, unsophisticated, and filled with blind posturing, they should be unstructured enough to let us exploit some time-honored text processing techniques to fake our way through them. In particular, we can easily build a very fast system that:

- Recognizes good and bad events in the game (its team winning, its team losing, killing, getting killed, etc.), and reacts with appropriate responses from a very large repertoire.
- Roughly recognizes when other players are trash-talking, and responds appropriately depending on whether the player is a friend, a foe, a victor, etc.
- Ignores anything too complex to process, such as unrelated bits of conversation about real-life events.

The simplicity of the conversation aids the bot in disguising itself. Human players will not likely be overly suspicious if the bot misses some remarks directed towards it, or makes nonsensical comments from time to time. Furthermore, the presence of obvious stereotypes provides a clear guide for development.

Some first-person shooters, such as Quake Arena, have attempted to produce some chatter from bots during games, usually in the form of taunts. However, action games rarely allow for leisurely typing time during the game itself. Rather, most chatting is done outside actual game play. In fact, over-chatting during a game itself can quickly unmask the bot.

Sample implementation

We have constructed a system that, guided by the above observations, attempts to engage in social interaction with human players. The following is a brief description.

Counter-Strike

As our testbed and development platform, we selected the first-person shooter Counter-Strike. This is a popular game, with an average of twenty thousand online players and over four thousand active game servers at any time. The game consists of short scenarios of four minutes or less. When a player is eliminated, he remains in the game as a spectator until the next scenario begins. Eliminated players often observe the ongoing action and chat with one another extensively. Hence, the game mechanics are tailor-made for our experiment. We have a large population of juveniles playing the game, many of whom fit the stereotypes we mentioned in the previous section, and the game itself includes periods of inactivity where the players are implicitly encouraged to communicate with one another through a chat interface.

Although Counter-Strike was originally developed as a purely multi-player online game, there exists a number of bots developed by members of the Counter-Strike community. These bots are readily available for download, often including source code. We chose one of these bots, named Teambot (http://www.teambot.net), as the foundation for our experiment.

The original Teambot had some chatting capability, but only had a small preset number of canned responses. Moreover, the NPC would only chat very sporadically. We decided to remove all of the original chatting code in Teambot, and implement our own version.

Architecture

Figure 1 shows an outline of our architecture design.

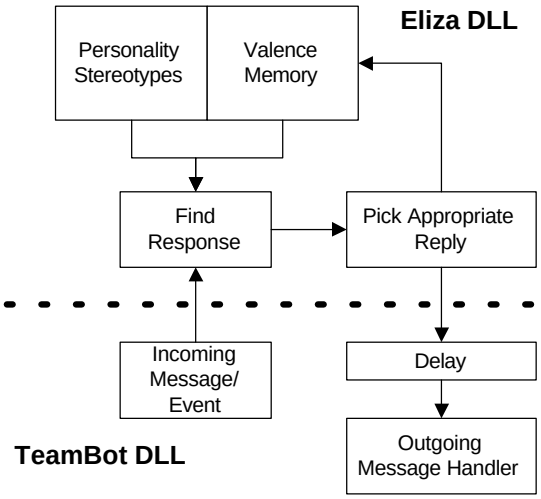

Figure 1 : TeamBot Chat Interface

The fast system that responds to what goes on in the game is essentially an augmented version of an Eliza. It is just a simple state-machine-based text pattern-matcher, written in C for extra speed.

Incoming chat messages or significant game events are sent to Eliza. The Eliza program is equipped with a set of stereotypical personalities, including 'boaster', 'whiner', and 'warez dude' – each of these personalities includes triggers for matching events and conversational bits that would be interesting to the particular personality. For example, 'warez dude' would be interested in talking

about pirated software, ‘boaster’ will have an extended repertoire of put-downs and boasts, and so on. The bot will use one particular personality for the duration of a game. Within each personality is a different priority stack of speech behaviors. The incoming messages or events are text-based, and are matched against regular expressions within each behavior. The first matching behavior will generate a response. At the bottom of each priority stack is a confusion script behavior, which generates generic stock responses to inputs that seem important but cannot be discerned by any other behavior.

Not every input message and event is passed to the Eliza module. The bot determines probabilistically whether or not it will respond to a given message or event. When it is actively participating in the game, the bot has a very low probability of responding to any messages. After it has been eliminated from the current game and is merely observing, the bot will become far more responsive.

Furthermore, the bot maintains a simple valence memory of how it feels towards each of the players currently in the game, and this valence influences the bot’s responses. Valence is responsible for maintaining some semblance of context in the bot’s conversations. A high positive value indicates that the bot is positively predisposed towards the player, while a high negative value shows severe disdain. Each response has a valence modifier attached. When a response is chosen by the Eliza module, the modifier is used to adjust the valence value for the appropriate player.

The output response chosen is delayed by a small amount of time before being passed to the outgoing message handler. This prevents the bot from producing output faster than humanly possible.

Results

We started a game server online for sixteen players and staffed eight of those slots with our modified Teambot NPC. From a tactical standpoint, the bots were somewhat competent on the maps on which waypoints had been prepared to aid with navigation, although they still performed some strange maneuvers occasionally. We logged all conversations that occurred within the game, including any between the bots and human players. Some of the more interesting exchanges are presented at the end of this paper

Our main objective was to observe how far we could push a simple mechanism like an augmented Eliza to fooling a human into failing the Turing test. We found that, in general, we could fool some of the humans for a good amount of time, but there were people who caught on surprisingly quickly. We have some idea why.

The primary fault we found with the current system is that Eliza-based conversations are too schizophrenic to be believable over a long period of time. The bots would jump from topic to topic without any history other than the general valence towards the speaker. The system needs a better way of representing the current topic, and needs to maintain a minimal topic history. A mechanism such as a decaying episodic memory model associated with each player in the game would be in order.

Another important problem was stylistic, concerned with the extraction of the ‘addressee’ of messages specifically targeted towards particular players. Rather than referring to a person by their full name, players generally use some abbreviation, for example, ‘robzilla the horrible’ would be referred to simply as ‘robzilla’, or even ‘rob’. The bots had no understanding of this, so they routinely missed messages directed specifically at them. Conversely, when referring to a player, the bots would utilize the player’s full name. This appears very strange particularly when online players have a habit of using odd symbols as part of their name or have very long names. For example, ‘@Zbk@$hooter!’ or ‘Bill Nye the Violence Guy’. This reduced the realism of the bots and aided players in breaking through the illusion.

Finally, there was a serious, but purely technical glitch. All players within a multi-player game server show a specific round-trip ping time to the server. Because they sit directly on the server, bots show up as having a ping time of only 5ms. This is far too low for a real human player and quickly reveals the player is a bot rather than a human.

Conclusions

Our hypothesis is that the intelligence or skill of an opponent in a computer game matters less than the social interaction that arises in a multi-player scenario. Furthermore, we submit that it is possible to replicate many interesting online conversational patterns through the use of simple mechanisms, without requiring full-blown natural language comprehension. Our experience with the Eliza-based system in Counter-Strike has shown us that this approach has potential. Certainly our NPCs managed to deceive most human players for a decent amount of time. However, over the long term, the simplistic responses unmask the illusion for what it is. We have identified specific points of failure in this preliminary experiment, and will be conducting tests on a more sophisticated chatting NPC in the future.

Acknowledgements

We would like to express our gratitude to the Teambot developer, Alistair Stewart, and the Counter-Strike mod community for the resources and help they kindly provided.

References

Jenkins, H. 1998. Complete Freedom of Movement: Video Games as Gendered Play Spaces. In *From Barbie to Mortal Kombat: Gender and Computer Games*. Cambridge, MA: MIT Press.

Laird, J. E. 2000. It Knows What You're Going To Do : Adding anticipation to a QuakeBot. *AAAI 2000 Spring Symposium on Artificial Intelligence and Interactive Entertainment.* AAAI Technical Report SS00–02. Menlo Park, CA: AAAI Press.

Laird, J. E. and van Lent, M. 1999. Developing an Artificial Intelligence Engine. In *Proceedings of the Game Developers Conference*, 577-588. San Jose, CA.

Appendix: excerpts from game logs

The following are a few amusing excerpts from recorded in-game conversations. Bots' names are printed in **boldface**.

Example conversation with bots

***DEAD*!PRV!AKOrbValk** : whats the name of this map
***DEAD*@PRV@DeathFubar** : guard that bomb
*DEAD*Dr-Azrael : hah
*DEAD*Silent Bob : we black
***DEAD*+EwokAce-KEC+** : doh
*DEAD*Dr-Azrael : de_dust *(← responds with the name of the map)*
***DEAD*EvilSuperMalachi** : i'm sick of being capped
*DEAD*Silent Bob : *shakes head 'no'*
***DEAD*!hun!EvilSuperFoo** : how do you use the scope?
*DEAD*Dr-Azrael : right click

Accord answering a bot, and catching that

***DEAD*@kNP@SecretToolValk** : who are the bots here?
Accord : the people with 5 ping are bots
FunkyFunk : i hear the 5 ping thing is a server problem
BossWhax : heard anything about the new cs expansion?
Accord : damnit, i just answered a bot
PsychoPainHead : i've known damnit, i just answered forever, he aint no bot

Silent Bob agreeing with bots, Dr-Azrael answering

Dr-Azrael : who me?
Dr-Azrael : damn it i was reloading
CountFoo[NUCS] : real men dont camp
@MonsterFooQuee-NUCS@ : whip that f--- a--
Silent Bob : *nods*
Dr-Azrael : i wasn't camping