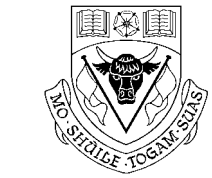

UNIVERSITY OF
CALGARY

Massive

Multiple Agent Simulation System in a Virtual Environment

Guy Davis
253194

SENG 609.22
Dr. B. H. Far
02/01/03

Abstract

A continuing challenge for special-effects creators in today's films is the generation of realistic behaviour from digital crowds. When the scope of a shot far out paces anything achievable with live actors, computer-generated (CG) actors must fill the void. To date however, most CG crowds often lack the realistic motion or individual behaviour required to convince audiences of their realism. In the "Lord of the Rings" film trilogy, the battle scenes ,with as many as 100 000 warriors, are animated using a tool called Massive. Combining motion capture photography, artificial intelligence, and agent-based software design, the Massive system endows each CG actor with dynamic and unpredictable behaviour. By allowing for emergent agent behaviour, the overall battle scene is significantly more realistic and impressive than previous approaches.

Introduction

For film makers today, the use of computer-generated graphics has become an essential tool for creating shots that simply could not be done in the real world. By using advanced animation techniques, fantastical creatures and landscapes can created to amaze audiences. The challenge for animators is to ensure that their digital creations are convincing to viewers.

One area of particular difficulty to create convincing effects for is the use of digital crowds. The audience is very perceptive at noticing small deviations from normal behaviour or motion by participants in a crowd. As well, the overall motions of the crowd must appear life-like in order to provide the level of realism required.

Recently, the practises of agent-based software engineering were applied to this area of crowd animation for the battle scenes within the "Lord of the Rings" movie trilogy currently in theatres. An agent-based approach was chosen as the developers felt that the " believability of a cast of thousands depends on the actions of individuals." [3] The software used to accomplish this feat, known as Massive, used individual agents with their own artificial intelligence to create the emergent behaviour required to convincingly portray a mythic battle scene.

Previous Approaches

Before Massive, most animators used a particle physics approach to animating large crowds of individuals. Also known as flocking, the animator utilized the basic forces of attraction and repulsion between individual particles (one per character) to simulate the dynamic interactions taking place within a crowd. [3]

The main drawback to this approach is that people's motions don't necessary follow pool-ball trajectories. Human movement is much more complex than simple Newtonian physics. Compounding this problem, is the ability all people have for recognizing actual human movement and discerning abnormalities from the norm. Basically the difficulty for animators is that the audiences have a keen sense of non-natural human motion.

Another approach currently used for computer-generated crowd dynamics by the film industry is known as copy and pasting. This approach realizes that filming actual actors gives a much more life-like feel to the crowd's movement than a crowd based on particle physics. Unfortunately, it is often not

feasible to obtain enough human extras or building facilities to accommodate a real shot. A good example of this was the scene of the crowd, numbering in the tens of thousands, in the ancient Roman Coliseum for the movie "Gladiator". To achieve the crowd for the coliseum scenes, a group of 200 human extras were filmed. This group of live people were digitally altered and copied throughout the entire coliseum in order to give the illusion of a full stadium. While each small section of the crowd appears realistic, the overall affect can be disconcerting as each section is nearly a perfect duplicate of every other section.[5]

Clearly, the primary goal of crowd and battle scene animators is to generate life-like behaviour from a large number of computer-generated characters without reverting to either copy/paste or flocking. Ideally, each CG character would have inherent, non-deterministic behaviour of their own. If the behaviour of each character is both convincing and distinct, then the behaviour of the overall crowd will very realistic.

Agent-based Computing

An agent can be defined as an "encapsulated computer system, situated in some environment, and capable of autonomous action in that environment in order to meet its design goals".[2] Some of the attributes which an agent might possess include:[1]

- Reactivity
- Autonomy
- Collaboration
- Knowledge-level communication ability
- Inferential capability
- Temporal continuity
- Personality
- Adaptability
- Mobility

A synthesis of work in the fields of artificial intelligence, software engineering, and operation research; agent-based computing provides an alternative to traditional computer science approaches to tackling complexity in the problem domain. By avoiding a top-down approach to control of complex systems, agent-based systems can often succeed using emergent behaviour.

An Overview of Massive

During the initial planning stages for Lord of the Rings film trilogy, the director, Peter Jackson, indicated that to properly transfer the battle scenes that J. R. R. Tolkien had envisioned to the medium of film would require battles involving over 50 000 participants.[5] Of course, it wasn't feasible to find, outfit, and then choreograph that many human extras in a live battle scene. The only solution was to use computer-generated graphics.

The challenge was given to Stephen Regelous, working with WETA Digital from New Zealand, to

generate realistic crowd and battle scenes involving thousands of CG characters. Starting in 1995, Regelous began development on an ambitious project to overcome the previous drawbacks inherent in the crowd animation approaches discussed above. Originally designed for the battle scenes, Massive was extended to create digital doubles for human actors as well as simulating a flock of crows.[6]

Massive Agents

A key strength of Massive is that it provides emergent behaviour for each CG character in scenes involving many CG participants. To accomplish this feat, Regelous adopted an agent-based approach to the Massive development effort. In Massive, each character is a distinct agent with its own quirks and personality traits. While the animators can use Massive to generally direct the behaviour of a character, the actual observed actions are not deterministic.

This element of surprise inherent in each new run of the simulation was one of the joys of Massive as described by director Peter Jackson,

> "We literally didn't know what these guys were going to do. I still get a chuckle when I think about one of the first tests, where we had about 2000 foreground guys desperately trying to kill each other; but in the background, about fifty of them had thought the better of it and had turned around to flee the battle! I thought, 'Those are the smart guys.' It was extraordinarily spooky."[8]

Each character within Massive was provided with sensory inputs from which it makes decisions based its “genetic makeup”, personality controls set by the animators. The agents were given the ability to perform a range motions and actions in order to interact with their virtual environment.[5]

Sensory Inputs

A primary input for Massive agents is their sense of sight. Each agent receives a scan-line rendered image of the scene around him for every frame of the film. By providing agents with a visual image, Regelous was able to avoid the impossible task of having each agent search the entire scene database. Using the image, agents could concentrate only other agents and obstacles that were within their visible range, safely ignoring agents miles away. As well, this use of a scanned image is closer to the imperfect information that real people are given by their vision.[8]

Massive agents also have a rudimentary sense of hearing to provide further information regarding their environment. Using a simulation of physical sound, each agent would output a certain “pitch” corresponding to their type, such as elf, orc, human, or dwarf. As agents were essentially “humming” all the time, other agents could be aware of agents near them without necessarily seeing them. This was primarily used to avoid collisions during agent locomotion. [8]

A crude sense of touch was also provided to the agents through collision detection. This sensory input allowed agents to respond to collisions with other agents and with the terrain of digital environment. This proved useful for enabling realistic interaction with rough terrain such as rocks, streams, trees, and hills. [8]

Agent Actions

To provide lifelike actions, each agent type was provided with a library over 350 different movements derived from motion-capture sessions with real actors.[5] This approach yielded over thirty distinct

agent prototypes based on character and weapon such as “orc battleax” or “elf sword”. Each agent prototype then was configured for various parameters such as leg length, arm thickness, strength, and aggressiveness. Each variable included an upper and lower limit. The use of a range would then allow for random variation within limits for all agent instances created from these prototypes.[8]

By instantiating a large number of each prototype agent, a battle scene could be created that would still ensure individual behaviour. In fact, each agent was choosing an action from its extensive list of motion captures 24 times every second. For example, an agent might have a choice of six different sword strokes depending on whether it was moving forward or raising his shield. [8]

Artificial Intelligence

In order to ensure realistic behaviour from each character, a reasonable level of artificial intelligence was needed to ensure that lifelike responses to the environment were selected. To achieve this, Regelous designed between 100-8000 behavioural logic nodes for each agent that allowed them to interpret, analyze, and respond to their environment. By grouping these nodes into rule clusters that govern movement across varied terrain, aggression, fighting style and many other factors, it was possible to fine tune the behaviour of an agent prototype.

Drawing upon the work of Karl Sims, Massive employs a language of nodes and connections as its primitive elements to describe directed graphs representing the neural circuitry of its agents.[7] While Massive did not explicitly use genetic algorithms as advocated by Sims, it did evolve the characters through iterative development and improvement by the animators.

Additionally, fuzzy logic was employed to allow for more descriptive behavioural control. Given the difficulty of describing complex systems in absolute terms, fuzzy logic is a departure from classical two-valued sets and logic, that uses "soft" linguistic (e.g. large, hot, tall) system variables and a continuous range of truth values in the interval [0,1], rather than strict binary (True or False) decisions and assignments.[10]

Initially the control of a character's directed graph was done with pen and paper. However, this rapidly become an inefficient approach as the characters grew in complexity. Eventually, a custom tool was written to allow drag and drop editing of the connections forming a character's brain. This tool, combined with powerful render farms of Linux computers, allowed the film's director to actually direct the computer-generated characters. The animators could quickly rework the behavioural controls of characters to better suit the individual scenes.[8]

Conclusion

The Massive software tool presents a novel use of agent-based software engineering techniques to tackle the problem of designing convincing virtual creatures. Avoiding previous top-down control methods employed by computer-generated crowd simulators, Massive is able to generate significantly more lifelike behaviour through emergent agent action.

Each Massive agent is a composite of hundreds of motion capture images, a large set of behaviour controls calibrated by the designers, and a number of fuzzy logic decision nodes. This combination ensured that the actions of each individual agent were realistic and non-deterministic. By ensuring the convincing behaviour of each agent, the behaviour of the crowd or battle scene was ensured.

These scenes are currently winning rave reviews in with the release of “Lord of the Rings: The Two Towers”, so it is clear that the application of agent-based software engineering techniques to this

problem has dramatically improved the state of the art in computer-generated crowd simulation.

References

1.Far, B. H. SENG 609.22 Session 1 Class Handouts. 2002.
http://www.enel.ucalgary.ca/People/far/Lectures/SENG609-22/PDF/SENG609-22-01_handout.pdf
2.Jennings, Nick. Agent-based Computing: Promise and Perils. 2002.
http://www.ecs.soton.ac.uk/~nrj/download-files/computers-thought.pdf
3.Koeppel, Dan. Massive Attack. Popular Science Magazine. 2002.
http://www.popsci.com/popsci/science/article/0,12543,390918-1,00.html
4.Lehane, Scott. The Lord of the Rings. Film and Video Magazine. 2001.
http://www.filmandvideomagazine.com/2001/10_oct/features/lordorings_oct.htm
5.Macavinta, Courtney. Digital Actors in Rings Can Think. Wired Magazine. 2002.
http://www.wired.com/news/digiwood/0,1412,56778,00.html?tw=wn_ascii
6.Robertson, Barbara. The Fellowship of the Ring. Computer Graphics World Magazine. 2002.
http://web.umr.edu/~dsistler/jan03/articles/computerrelated/lotr-cgw.pdf
7.Sims, Karl. Evolving Virtual Creatures. Computer Graphics (Siggraph '94 Proceedings), July 1994, pp.15-22.
8.Thomas, Christopher. Lord of the Rings: The Prologue Battle. University of Georgia. 2002
http://www.arches.uga.edu/~chaos/presentation/slides/lotr_prologue_battle.html
9.Tonde, Allan. Fuzzy Logic – The Basics. GTE Government Systems Corp. 2002.
http://www.austinlinks.com/Fuzzy/basics.html