Human Behavior Models and Unreal Tournament

Soar Workshop
June 27, 2003

Ryan McAlinden
Institute for Creative Technologies
mcalindend@ict.usc.edu
Introduction

Objective: Demonstration of simulation-independent, goal-directed tactical Human Behavior Models (HBMs) that reflect Non-Player Character (NPC) behavior, navigation, and emotion

Team
- Institute for Creative Technologies (ICT)—Lead software integrator
- University of Michigan—Soar
- Biographic Technologies—AI.Implant
- University of Pennsylvania—Performance Moderator Functions (PMFs)
- Quicksilver Software, Inc—Unreal Tournament assets
- Defense Modeling and Simulation Organization (DMSO)

Component Overview
- Simulation Environment—Unreal Tournament
- 3 individually developed Human Behavior Models (HBMs)
  - Soar: serves as the central behavior generation component
  - AI.Implant: Path-planning and navigation middleware
  - Performance Moderator Functions: Emotional and physiological effect’s modeling
- Customized Unreal Tournament assets and scripts
  - ICT
  - Quicksilver Software, Inc.
HBM Architecture

- **SoarBots**: SGIO
- **AllBots**: All Data Manager
- **PMFBots**: PMF Data Manager

Native functions:
- Army Ranger Subordinates
- Civilians/combatants
- Civilians

Interfaces:
- SoarInterface
- AllInterface
- PMFInterface

UScript/C++ Interface

COM
Unreal Tournament—Why?

- Highly modularized and replaceable
- A C++ interface based on an object model that is similar to Microsoft Foundation Classes (MFC)
- Supports dynamic loading of DLLs and scripts on demand
- Robust debugging environment, with Visual C++ debugger support
- An UnrealScript interface based on an object model that is similar to Java
Soar

- Utilizes existing Agent MOUTBots—added follow and tactical scanning operators
  - Manages subordinate bot navigation, communication, and attack behaviors

Nuggets
- Straightforward operator/production additions
- SGIO
- n-level behavior modeling

Coal
- Difficult to debug
- Communication between Soar and the Game Engine
Overview
- Autonomous character creation and management – allows developers to build and control Game AI more easily
  - Group/targeted behaviors
  - Basic navigation and path planning
- SDK – C++ middleware residing from the Game Logic Layer (state of the simulation) to the Physics layer

Nuggets
- Simple development of complex AI controls
- Developer extendibility of existing behaviors
- Relatively simulation-independent
- Straightforward SDK that contains a suite of C++ APIs

Coal
- Limited set of included behaviors
- Isolated APIs
- No platform cross-compatibility
- Documentation
Performance Moderator Functions (PMFs)

- Reflects physiological and emotional states of simulation-controlled players

**Nuggets**
- Modular system that is simulation-independent
- Models multiple “coping styles” within one architecture
- Multiple levels of CGF customization
- Reusable, rapidly composable entities

**Coal**
- Lacks a central AI system to control other aspects of character intelligence, movement, and behavior

Diagram:
- Physiology
  - Stress
  - Coping Style
  - Perception
  - Emotion/Culture
  - Emotional Utility
  - Decision*
HBMs can enhance current simulation capabilities by modeling fairly complex human behaviors and emotions catered to a specific discipline or genre (i.e. military tactics)

- Able to qualitatively evaluate how different behavior models interact with one another simultaneously
- Unreal I Game Engine provides a relatively seamless integration of external components
- Relatively robust communication mechanisms of HBMs

- Too tightly integrated with simulation
- Game Engine and asynchronous HBMs don’t always exchange data correctly
Demonstration

- Custom-developed Mogadishu level, animations and character models
  - Based off *Black Hawk Down* scenario
- 3 subordinate SoarBots (Army Rangers)
- 2-5 AI.Implant Bots (civilians, militia)
- 1 PMF Bot

Rapidly Deployable, PMF-Based Human Behavior Modeling

PMFServ

SOAR

AI-Implant ™
Questions?