Attribute Learning System

[Applying Genetic Algorithms to Improve RPG Combat Mechanics]

Austin Cory Bart
Virginia Tech
acbart@vt.edu

K. Alnajar
Virginia Tech
kar@vt.edu

ABSTRACT
Having a good set of moves for players to choose from in role-playing games (RPG) is essential for the game to succeed. Often times in an RPG, the players have various attributes which these moves can effect and coming up with good formulas for this is not easy. The process of creating an effective set of moves can take time and can be a difficult challenge to overcome in the design process. This paper propeses an implementation to effectively create these moves using a genetic algorithm implementation. Two seperate implementation styles of genetic algorithms are used, a tree style and a vector style. The results show that the vector styled approach for the genetic algorithm shows promising results in move set creation.

Categories and Subject Descriptors
1.2.1 [**ARTIFICIAL INTELLIGENCE**]: Applications and Expert Systems — *games*

General Terms
Genetic Programming

Keywords
Genetic, Programming, Game, Development

1. PROBLEM
RPG combat mechanics - attributes (primary vs. secondary)

2. APPROACH
2.1 Prior Work
2.2 Target Audience
3. IMPLEMENTATION
3.1 Genetic Algorithm
General description of a fitness function

3.2 Fitness Function
3.2.1 Simulation
Tried several approaches:

- Move usage**
- Battle victory**
- Battle length
- Linearity

3.3 Function Tree
3.3.1 Mutation Algorithm

Mutating entire tree

Mutating leaf nodes

3.3.2 Cross-over Algorithm

Uniform cross-over

3.4 Function Vector
Modeled as vector of coeffecients

3.4.1 Mutation Algorithm
3.4.2 Cross-over Algorithm
3.5 Players
3.5.1 Minimax Player
3.5.2 Greedy Player
3.5.3 Random Player
3.5.4 Player Utility calculation
4. EXPERIMENTS
4.1 Validation of Genetic Operators
4.1.1 Function Tree

Mutation. Levenshtein Edit Distance and Numerical Analysis

Cross-over. Levenshtein Edit Distance and Numerical Analysis

4.1.2 Function Vector

Mutation. Levenshtein Edit Distance and Numerical Analysis

Cross-over. Levenshtein Edit Distance and Numerical Analysis

4.2 Move Generation Experiments

4.2.1 Tree vs. Vector

4.2.2 Mutation Rate vs. Cross-over Rate vs. Parents Retained

4.2.3 Attribute Type Frequencies

4.3 Validation of Moves

Ran simulations of Minimax/Greedy/Random to determine whether Minimax dominated.

5. CONCLUSION

6. FUTURE WORK

6.1 Genetic Operators

7. REFERENCES

8. REFERENCES