Title: Simulation of intelligent behavior of Non-Player Characters in First-Person Shooter computer game environment using Machine Learning techniques.

1. Introduction
   1. ??? – leave it for later
2. Background
   1. Artificial Intelligence in First-Person Shooter games
      1. Introduction
      2. Bots architecture
      3. Navigation solutions
      4. Finite State Machines
      5. Fuzzy Logic
      6. Scripting
   2. Machine learning in Computer Games (Leave it? Remove it?)
      1. Introduction
      2. Online and offline learning
      3. Testing issues
   3. Optimization methods
      1. Introduction
      2. ???
      3. Popular optimization algorithms
      4. Stochastic optimization
   4. Reinforcement Learning
      1. ??? – more information
   5. Quake II and QASE API
      1. Introduction
      2. Game choice
      3. QASE API
      4. Related work
3. Project requirements (or: Thesis scope, or: Requirements and assumptions, or: Problem statement)
   1. Project scope (or: Project goal and scope? Or: The task)
      1. Designing and developing Quake II bot able to compete with human players and other third-party bots.
      2. Applying reinforcement learning for a selected problem
   2. Assumptions
      1. Implementing necessary minimum for effective navigation. Using prepared map knowledge.
      2. Focusing on combat module
   3. Limitations
      1. Client side – will have a delay
      2. Lack of full information about the world – elevators, gunshots, etc.
4. Developed solution (or: Methods)
   1. Reference bot
      1. How does making a bot look like in QASE
      2. Knowledge representation
         1. Waypoint Map – how is it created
         2. Enemy positions – updating them
         3. Seen entities
         4. Maybe other stuff from KB
      3. The loop
         1. Update knowledge base
         2. Establish navigation decision
         3. Get navigation instructions
         4. Establish firing decision
         5. Get firing instructions
         6. Execute instructions
      4. Navigation decision
         1. When we change it
         2. Spontaneous decisions
         3. Entity ranking – weights adjusted by hand
         4. Sending movement to server
      5. Firing decision
         1. Enemy and weapon choice – weights by hand
         2. Aiming algorithm
         3. Sending firing instructions
   2. Learn bot
      1. Reference bot + changing the configuration
   3. Bot’s launching and debugging application
      1. Launching
      2. Debugging
      3. Experiments ??
      4. Statistics
5. Experiments description
   1. Reinforcement learning ??
   2. Navigation optimization
      1. Experiment desired characteristics
      2. Choosing experiment procedure + justification
   3. Experiments application
      1. Algorithms chosen and why?
      2. GUI description
6. Results
   1. RL ? went bad?
   2. Algorithms comparison
   3. Performance of found solution
      1. In-game observations
      2. With Eraser bot
      3. With human players
7. Conclusions
8. Attachments
9. Bibliography