Artificial Intelligence

Institute of Computer Graphics and Algorithms

Vienna University of Technology

Artificial Intelligence

- The art of creating an immersive experience in regards of interaction and feedback, often creating a game challenge at the same time.
- Similar goals as physically correct simulation of your game environment :)
  - Similar design philosophy problems :>
  - And also a lot of new challenges :D

Basic Design Considerations

- What are your AI's objectives?
- How to accomplish that objectives / How should your AI behave ?
- What are the interfering factors between your AI and it's objectives ?
- How to exploit those well known factors to achieve the AI's goal?
  - How does your AI interact with it's environment?
- Sometimes simpler is better !
- There is no generic perfect solution

(The cake is a lie !)

Guidelines

- Your AI should be
  - Fun (It's not all about the technical aspects)
  - Interesting to explore
  - An appropriate mixture of random and foreseeable behaviour
  - Fitting the gamplay well
    - certain assumptions are possible

Concepts

- Finite State Machines
  - Multiple States
    - Idle
    - Walk
    - Attack
    - Die…
  - State Transitions
    - If an action is finished (e.g., after “attack” return to “idle”)
    - If an event occurs (e.g., actor is hit)
    - Add some random

Concepts

- Create different AI personalities
  - Condition:
    - Health, Speed
    - Favorite weapon
    - …
  - Parameters that influence state changes:
    - Aggressive – defensive (dodging)
    - Bold – cowardly (may run away when hit)
  - Add some random

Concepts

- Make your AI dumb
  - The „optimal“ AI is no fun to compete against
    - Use simpler problem solving approaches
      - performance and fairness at the same time
    - Implement reaction time for the actors
    - Give them a bad aim
      - E.g. only hit if you don‘t move
      - E.g. only hit if you move in a straight line

Decision making

- Greedy-Algorithm
  - Fast
  - Stable
  - Compute value for every possible situation
    - Decide for the situation with the highest value
    - Values represent the winning probability for a certain decision
  - Quality depends on the chosen heuristic to evaluate the value
  - Only reasonable in certain game scenarios

Decision making

- Minimax-Algorithm
  - 2 players, act alternately
    - Player 1 searches for maximum value in decision tree
    - Player 2 searches for minimum value
    - Values, again, represent the winning probability
  - Repeat recursively up to given depth, build tree
    - Decide for leaf with highest/lowest value
  - Also requires a heuristic, apparently

AI - Path Planning

- Needs graph structure
  - E.g., regular grid
  - Start nodes
  - Destination nodes
  - Edge costs
- Simplest solution:
  - Go straight to your enemy
- Other possibilities:
  - Dijkstra’s algorithm (shortest path)
  - Best-First-Search (heuristic approach)
  - A*-algorithm (combination of aforementioned)

AI – Performance considerations

- Exploit coherency, a lot of results can be reused
  - Group your actors
  - Often similar behaviour leads to even better results, chaotic behaviour tends to be distracting (~ not fun)
- Use the power of your view frustum!
  - Don't waste time on simulating things in a detailed manner, if the user can't see them anyway
  - Your CPU will be grateful

Further Information

- AI:
  - http://www.gamedev.net/reference/list.asp?categoryid=18
- Finite State Machine-Tutorial:
  - http://www.generation5.org/content/2003/FSM_Tutorial.asp
- Beginner’s guide:
  - http://ai-depot.com/Tutorial/PathFinding.html
- A* descriptions:
  - http://theory.stanford.edu/~amitp/GameProgramming/
  - http://en.wikipedia.org/wiki/A-star_search_algorithm
  - http://www.generation5.org/content/2000/astar.asp
- Intelligent path-finding
  - http://www.gamasutra.com/view/feature/3317/smart_move_intelligent_.php?page=1