Development of Games

Lecture 14

AI in Games

Outline

- What is AI
- Features of AI in Games

What is AI?

- What is intelligence (constrained to games)?
- When is a game considered intelligent?
- Do game agents need to pass the Turing test?

What is AI?

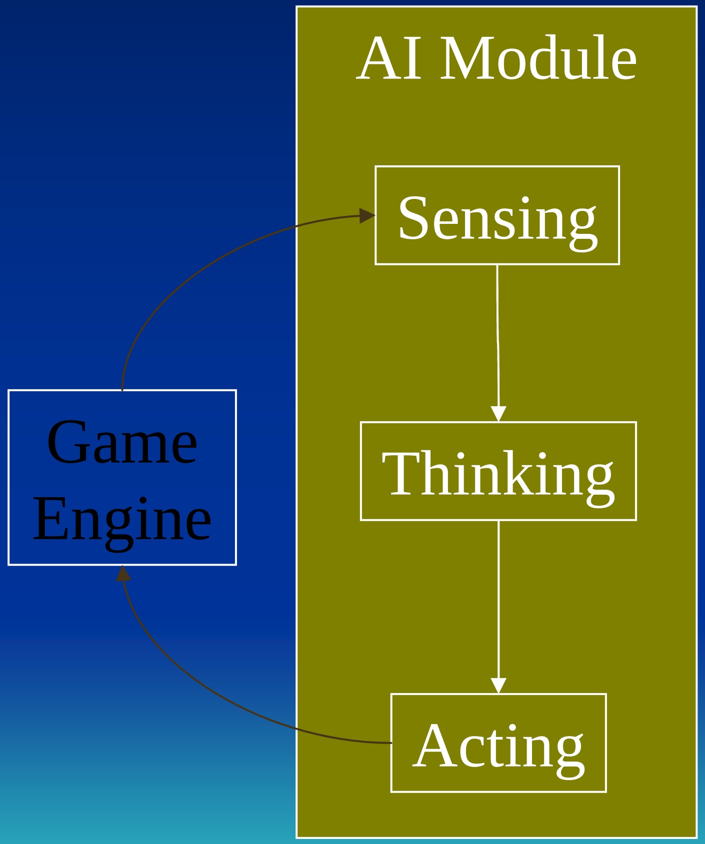
- AI is (traditionally) the control of every non-human entity in a game?
  - The other cars in a car game
  - The opponents and monsters in shooter games
  - Your units, your enemy's units and your enemy in a RTS game
- But, typically does not refer to passive things that just react to the player and never initiate action
  - That's physics or game logic
  - For example, the blocks in Tetris are not AI, nor is a flag blowing in the wind
  - It's a somewhat arbitrary distinction

Perception Management

1. Perceive (Sensor information)
2. Think (Evaluate perceived data)
3. Act (execute planned action)

AI Update Step

- The sensing phase determines the state of the world
  - May be very simple - state changes all come by message
  - Or complex - figure out what is visible, where your team is, etc
- The thinking phase decides what to do given the world
  - The core of AI
- The acting phase tells the animation what to do
  - Generally not that interesting



AI in the Game Loop

- AI is updated as part of the game loop, after user input, and before rendering
- There are issues here:
  - Which AI goes first?
  - Does the AI run on every frame?
  - Is the AI synchronized?

AI and Animation

- AI determines what to do and the animation does it
  - AI drives animation, deciding what action the animation system should be animating
  - Scenario 1: The AI issues orders like “move from A to B”, and it’s up to the animation system to do the rest
  - Scenario 2: The AI controls everything down to the animation clip to play
- Which scenario is best depends on the nature of the AI system and the nature of the animation system
  - Is the animation system based on move trees (motion capture), or physics, or something else
  - Does the AI look after collision avoidance? Does it do detailed planning?

AI by Polling

- The AI gets called at a fixed rate
- Senses: It looks to see what has changed in the world. For instance:
  - Queries what it can see
  - Checks to see if its animation has finished running
- And then acts on it
- Why is this generally inefficient?

Event Driven AI

- Event driven AI does everything in response to events in the world
  – Events sent by message (basically, a function gets called when a message arrives, just like a user interface)
- Example messages:
  – A certain amount of time has passed, so update yourself
  – You have heard a sound
  – Someone has entered your field of view
- Note that messages can completely replace sensing, but typically do not. Why not?
  – Real system are a mix - something changes, so you do some sensing

Requirements for Games AI

- They need to be fast (i.e. contrary to some academic AI approaches which are ‘optimal’)
- Games do not need optimal solutions (they need to be fun)
- Games may need to combine several techniques
- The goal is to design agents that provide the *illusion of intelligence*

Goals of Game AI

- Goal driven - the AI decides what it should do, and then figures out how to do it:
- Reactive - the AI responds immediately to changes in the world
- Knowledge intensive - the AI knows a lot about the world and how it behaves, and embodies knowledge in its own behavior
- Characteristic - Embodies a believable, consistent character
  - Fast and easy development
- Games do not need optimal solutions (they need to be fun)
- Games may need to combine several techniques
- The goal is to design agents that provide the *illusion of intelligenc*
- Low CPU and memory usage
- These conflict in almost every way – how?

AI Techniques in Games

- Basic problem: Given the state of the world, what should I do?
- A wide range of solutions in games:
  – Finite state machines, Decision trees, Rule based systems, Neural networks, Fuzzy logic
- A wider range of solutions in the academic world:
  – Complex planning systems, logic programming, genetic algorithms, Bayes-nets
  – Typically, too slow for games

Techniques used today

- Deterministic:
  - – Finite State Machines
  - – Decision trees
  - – Fuzzy State machines
- Statistical:
  - – Baysian Approaches
  - – Neural Networks
  - – Evolutionary algorithms/ Artificial life

Two Measures of Complexity

- Complexity of Execution
  - How fast does it run as more knowledge is added?
  - How much memory is required as more knowledge is added?
  - Determines the run-time cost of the AI
- Complexity of Specification
  - How hard is it to write the code?
  - As more “knowledge” is added, how much more code needs to be added?
  - Determines the development cost, and risk

Different games need different methods

- Action Games (lead a character through a set of levels)
- Fighting Games Physical combat
- Sports Games (icehockey and football)
- Racing Games (Car, bikes, speedboats)
- Adventure games (plots based on exploration and problem solving)
- Role Playing games (D&D)
- Board games (Chess, checkers)
- Strategic games (i.e. military type battle scenarios)
- Simulation Games (simcity)

Cases

- Chess: Large search trees
- Packman: Random FSM (weighted randomness)
- Smart environments: Information is not coded in the NPC but in their environment

General References

- AI and Computer Games
  - *Ian Millington. Artificial Intelligence for Games.* Elsevier, 2006.
  - www.gameai.com
  - www.gamedev.net
  - http://ai.eecs.umich.edu/people/laird/gamesresearch.html