IMGD 3000 - Technical Game Development I:
Intro to AI in Games

by
Robert W. Lindeman

gogo@wpi.edu
Motivation

- Okay, so you are in control
  - What about NPCs?

- Use AI to make your experience:
  - More compelling
  - More challenging

- Much AI is AS
  - Movement too simplistic
  - Movement too predictable
  - Movement too repeatable
Sample Uses of AI in Games

- Bad guys guarding something
- Bad guys looking for you
- Bad guys trying to beat you to something
- Bad guys trying to beat you (literally)
- Good guys working with you
- Other people just minding their own business

Summary: 4 Bad, 1 Good, 1 Neutral
More Detailed Examples

- **Bad guys**
  - Find a path through the environment from where they are to where you are
    - Pac man ghosts
  - Guard the base, but if I see you, then attack!
  - Opponents racing around a track
    - Be fast, but block you too

- **Good guys**
  - If we are attacked, then counter!

- **Neutral**
  - Act natural, please!
Let's Start Small

- Objects in an environment follow rules
  - Physical laws
  - Damage
  - Fatigue

- Particles
  - Water flows
  - Fire burns, rises, heat dissipates
  - These are just rules!

- Higher-order beings also follow rules
  - They are just more complex
Basic Model of Particle Systems

- A collection of many minute particles
- For each frame:
  - New particles are generated, and assigned a set of properties
  - Old particles die, and are removed
  - Remaining particles change their properties, e.g., position, shape, color
  - Objects are rendered based on this new state
- Creation and attribute manipulation are procedural
  - Can be the result of computations
Changing Particle Properties

How should the properties of the particles change over time?
- Where does each particle move to?
- How does its color change?

Can be based on *anything*
- Look at neighboring particles
- Look at scene objects, like obstacles
- Look at time
- Look at distance traveled
- Look at anything you want!
Basic Algorithm

Set up particle
While Animation In Progress
  If Particle Not Dead Then
    Add Particle Direction * Speed To Particle Position
    Add Particle Acceleration To Particle Speed
    Modify Particle Speed
    Modify Particle Energy
    If Particle’s Energy < Threshold Then
      Mark Particle As Dead
    End If
  If Particle Hits Object Then
    Modify Particles’ Positions, Directions, Speed and Energy
  End If
  Display Particle
End If
End While
Example: Movement of Particles

- $S_t$ is the state of all particles at time $t$
- At $t=0$: $S_0$

Images: Greg M. Johnson
(http://www.geocities.com/pterandon/boids.html)
Example: Movement of Particles

- Compute the influence of all other particles within some range
  - Attraction, repulsion

Images: Greg M. Johnson
(http://www.geocities.com/pterandon/boids.html)
Example: Movement of Particles

- Add all forces together, and use that to update the current position

Images: Greg M. Johnson
(http://www.geocities.com/pterandon/boids.html)
Example: Movement of Particles

- Wait, there might be other forces!
- Whatever the goal is of the scene

Images: Greg M. Johnson
(http://www.geocities.com/pterandon/boids.html)
Example: Movement of Particles

- Again, sum these as the forces on the particle
- Repeat these steps for each particle

Images: Greg M. Johnson
(http://www.geocities.com/pterandon/boids.html)
Particle Systems: More Examples

- Fire
- Explosions
Particle Systems: Final Thoughts

- In many cases, ignore self collisions
  - What does two fire particles colliding look like?

- Very general framework!
  - We can make special cases to get specific effects
  - Just change rules, objects, etc.

- How would you represent this system in code?