Flocking

Nat Guy

(Some slides borrowed from John See at Multimedia University, Malaysia)
Flocking

• Moving together in coordinated groups
• Birds in flocks, fish in schools, land animals in herds
• Murmuration of starlings:
  • https://www.youtube.com/watch?v=eakKfY5aHmY
Applications to Games

• NPCs can move in cohesive groups
  • Meadow of grazing sheep
  • Hunting flock of birds
  • Ants, bees, fish
• Other types of computer-controlled NPCs
  • Humans, orcs, catapults
  • Squadrons of aircraft
  • Friendly soldier squads
  • Crowds of people loitering
Behavioral Modeling of Flocking

• Craig Reynolds developed flocking model in 1986
• “Boids” model
• Presented at SIGGRAPH 1987: “Flocks, Herds, and Schools: A Distributed Behavioral Model”
• Later went on to do flocking animation for DreamWorks and Sony
Examples in Media

• First used for bats and penguins in Batman Returns (1992)
  • https://www.youtube.com/watch?v=Mo_1rAaj7FE#t=5
  • https://youtu.be/jCVwdeAobYc?t=15
• Jurassic Park (1993)
  • https://www.youtube.com/watch?v=nM-RPO10aPY
• Assassin’s Creed (various)
  • https://www.youtube.com/watch?v=ACWIRMepxk#t=597
• Countless other films and games
• Autonomous robotics:
  • GRASP Lab at UPenn:
    • https://www.youtube.com/watch?v=UQzuL60V9ng#t=27
Simple Rules of Flocking

- Leaderless flock of agents
- Each agent calculates its movements independently
- Agents can only see a few agents around them, their “neighborhood”
- 3 simple rules:
  - Cohesion
  - Alignment
  - Separation
Cohesion

- Each unit steers towards the average position of its neighbors.
- Units are attracted to one another as long as they are within range.
Alignment

- Each unit steers so as to align itself to the average heading of its neighbors
- Matches direction of units around it that it can detect
Separation

- Each unit steers to avoid hitting its neighbors
- Units are repelled by non-member units or obstacles. Repel effect can be inversely proportional to distance from unit
Mackerel “Baitball” Video

• https://www.youtube.com/watch?v=r1m6IkO26c#t=82
Neighborhood

- Range in which units can detect other units
Visibility

- Visibility constrained by field of view
- Also can be constrained by limited number of influencing neighbors
- Each unit is aware of its local surroundings
- Each unit does not necessarily know what the entire group is doing at any given time
Other Extensions

- Avoiding obstacles
- Avoiding predators
- Following leaders
- Making specific formations (circle, “flying V,” etc.)
Implementation

• In each game loop
  • Cycle through all units in the flock to acquire data (direction, speed, etc.) from unit’s neighbors
  • For each unit, update with net steering force from the three rules
• Each unit must update its list of current neighbors each game loop
Cohesion Implementation

• Calculate **average position** – vector sum of neighbors’ respective positions divided by total number of neighbors
• Determine direction to turn and angle to steer towards
• Steering force = (direction) * (steering force) * (angle of steering)
Alignment Implementation

• Calculate **average heading** – vector sum of neighbors’ respective alignments divided by total number of neighbors
• Determine direction to turn and angle to steer towards
• Steering force = (direction) * (steering force) * (angle of steering)
Separation Implementation

• Steer away from any neighbor that is within view AND within prescribed minimum separation distance (i.e., too close)

• Because this steering force is corrective, direction multiplier goes the opposite way

• Separation factor can be used to increase force with smaller separations

• Steering force = (direction) * (steering force) * (separation factor)
2D Flocking Demo (written by Nat in Python)

- [https://youtu.be/ipgdxQoVXWA](https://youtu.be/ipgdxQoVXWA)
- Note how the flocking pink agents respond to the presence of the single green agent, once it appears.
- (Sorry about the video capture quality!)
Further Resources

• Craig Reynolds’s Boids page
  • http://www.red3d.com/cwr/boids/