VGP332 – Artificial Intelligence

Instructor: Peter Chan

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

Assignment 4

- Questions?

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

Midterm

- No notes
- No computer
- No collaboration

- 1 hour

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

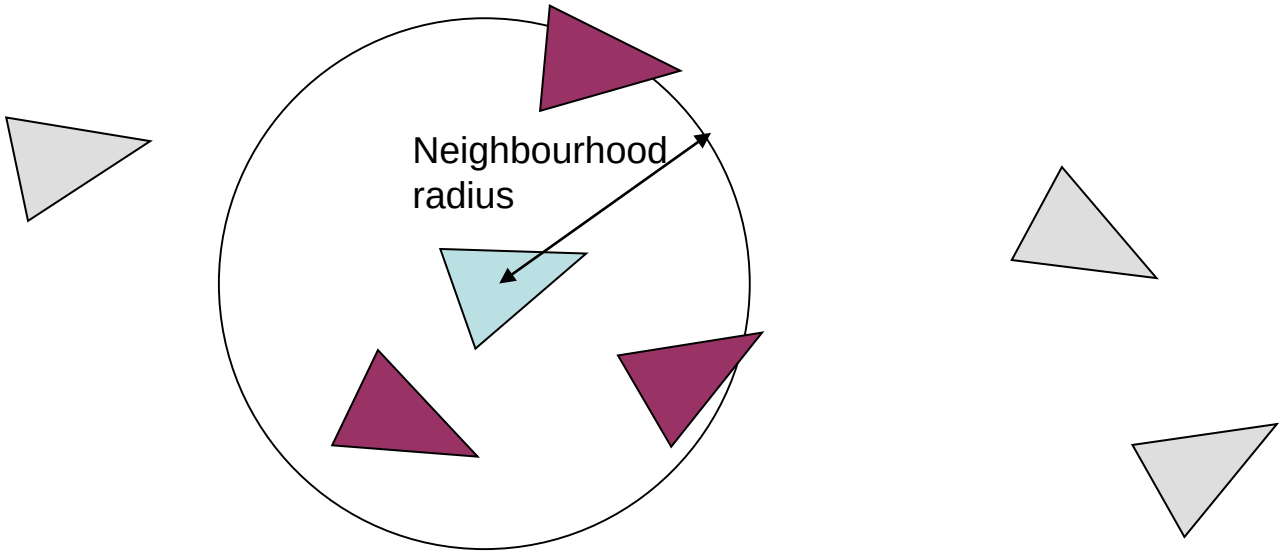
Group Behaviours

- Stanley & Stella

http://www.youtube.com/watch?v=pliaEEUzl0U

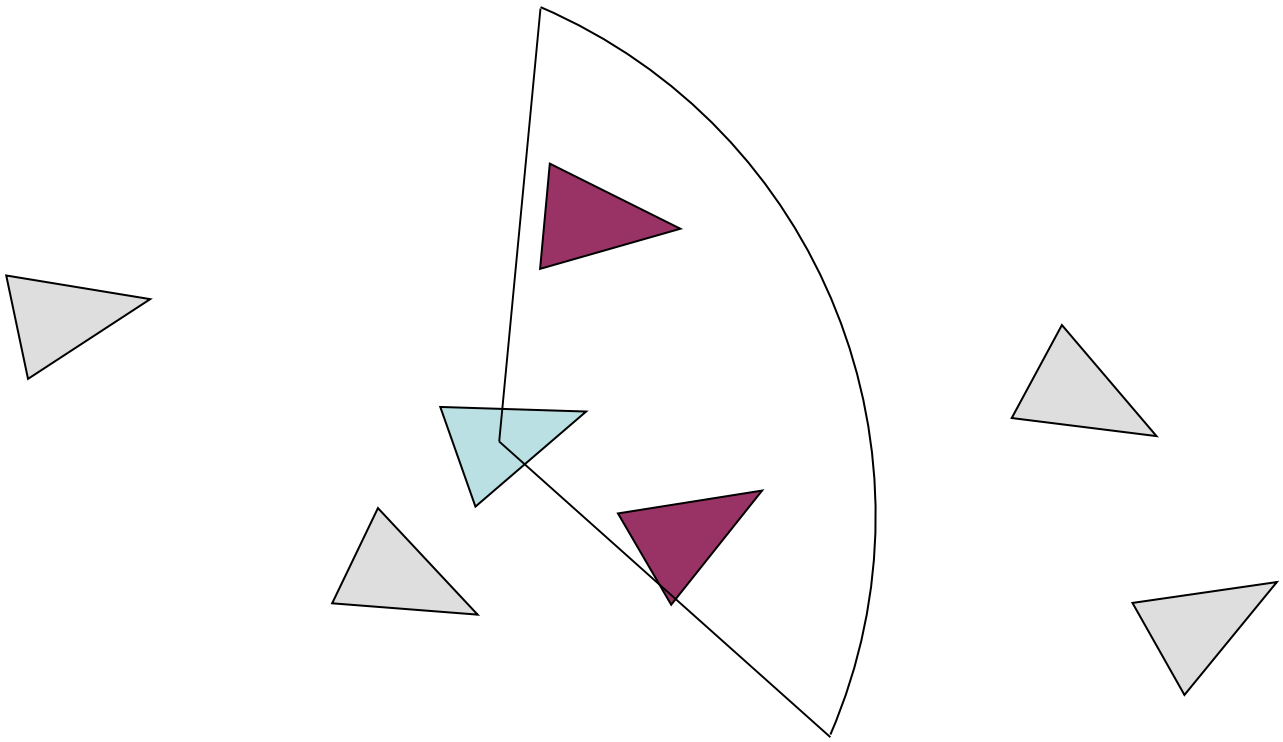
Group Behaviours

- To get group behaviours, an agent needs to be able to perceive other agents
- Simplification = only consider agents in neighbourhood radius



Group Behaviours

- More realistic = limited field of view



Group Behaviours

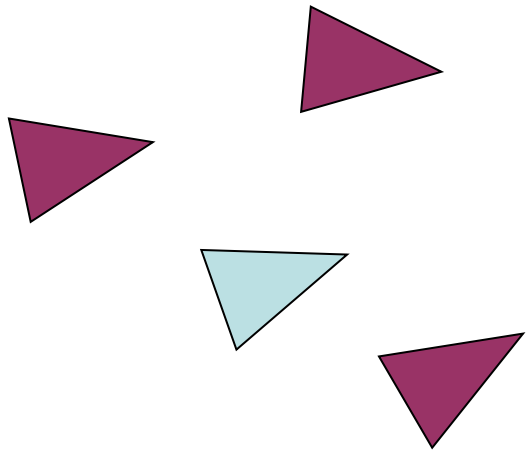
- Emergent behaviour
  - Individual agent follows specific behaviours
  - Collection of agents exhibit unexpected group behaviour
- Examples:
  - Flocking
  - Crickets' mating calls
  - Robot behaviour

Group Behaviours

- Separation
- Cohesion
- Alignment

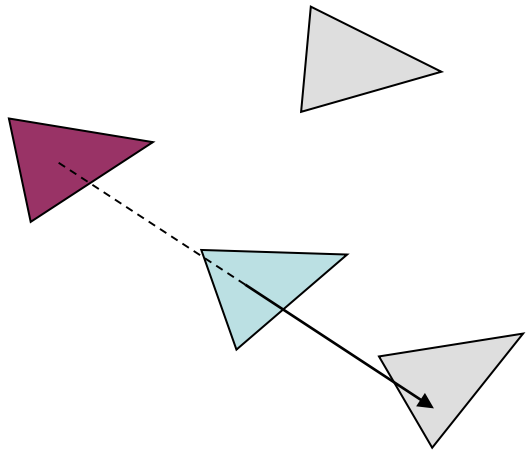
Separation

- Maximize distance from other agents in neighbourhood



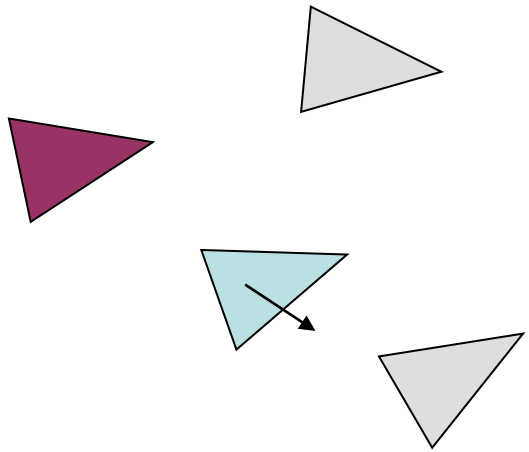
Separation

- Step 1: Determine vector away from neighbouring agent



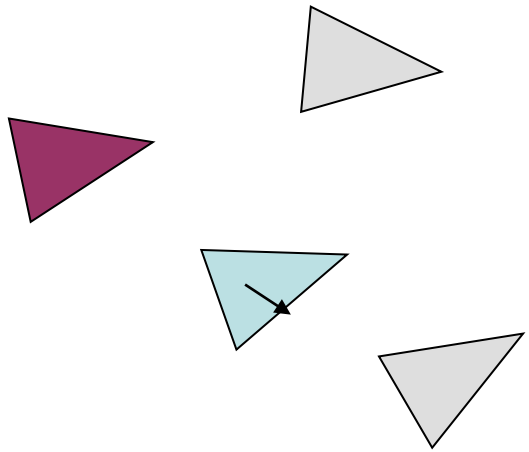
Separation

- Step 2: Normalize vector



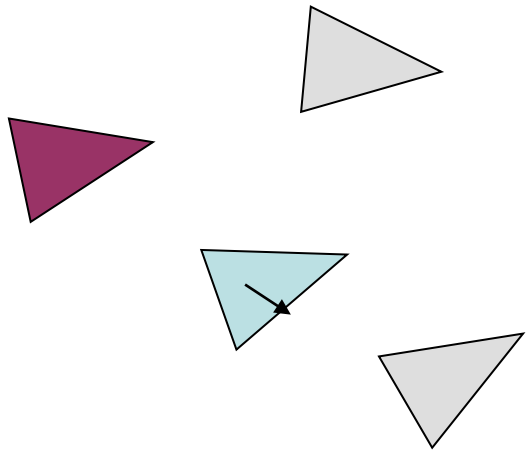
Separation

- Step 3: Divide by distance to neighbouring agent



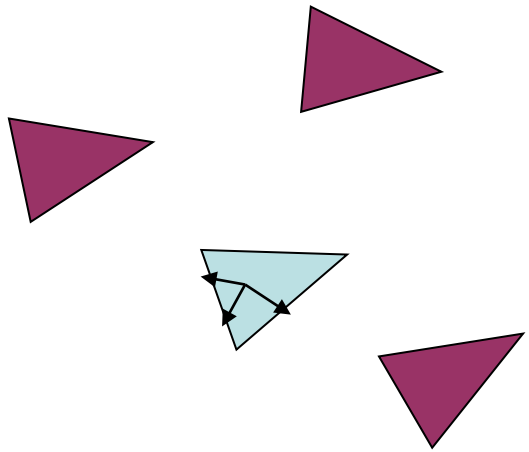
Separation

- Step 4: Accumulate in agent’s steering force



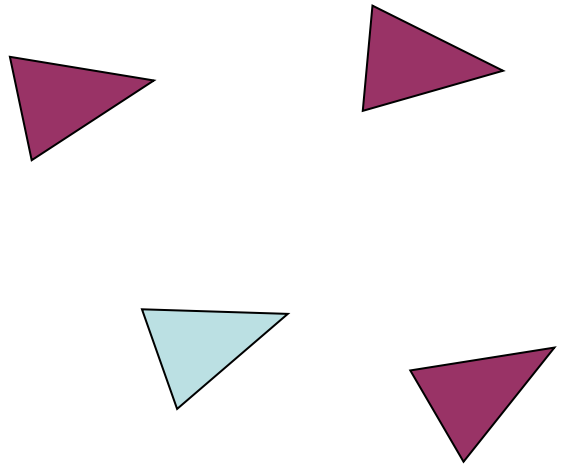
Separation

- Step 5: Repeat for all neighbouring agents



Separation

- Result = agents spread apart

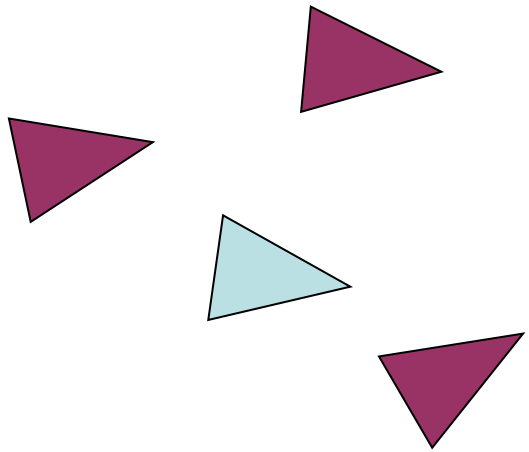


Separation

Look at code!

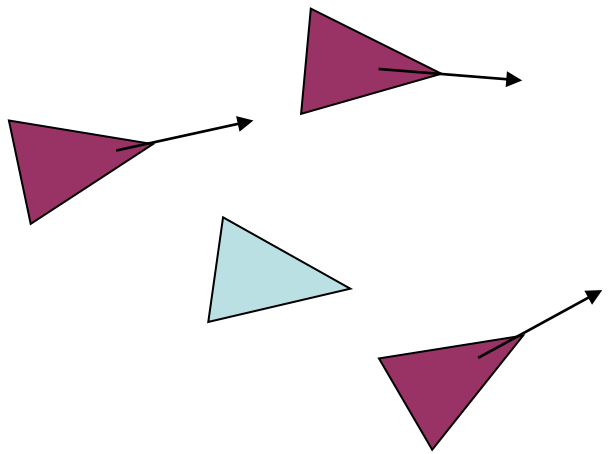
Alignment

- Keep agent’s heading aligned with neighbours



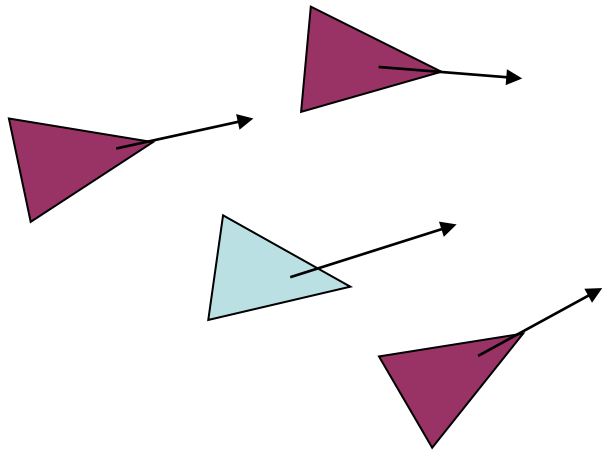
Alignment

- Step 1: Find heading of neighbouring agents



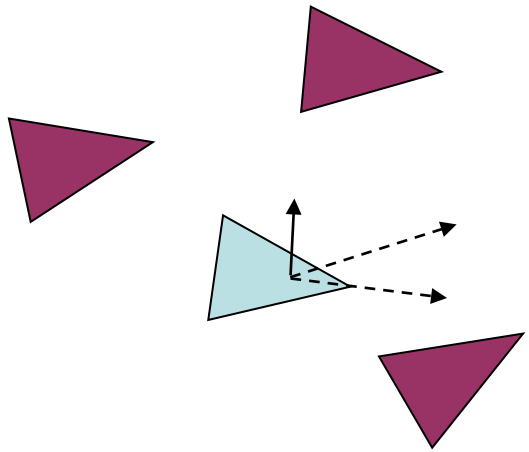
Alignment

- Step 2: Compute average heading



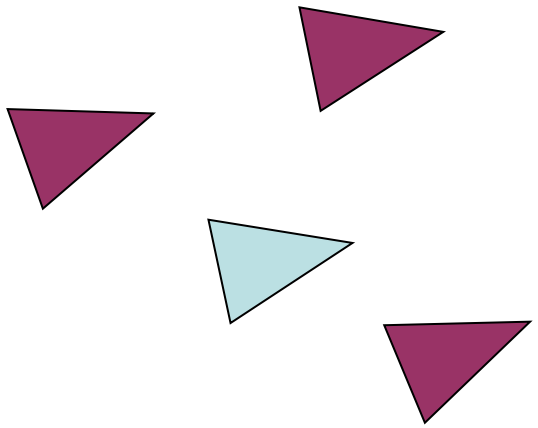
Alignment

- Step 3: Steering force =
  average heading - current heading



Alignment

- Result = agents align in same direction

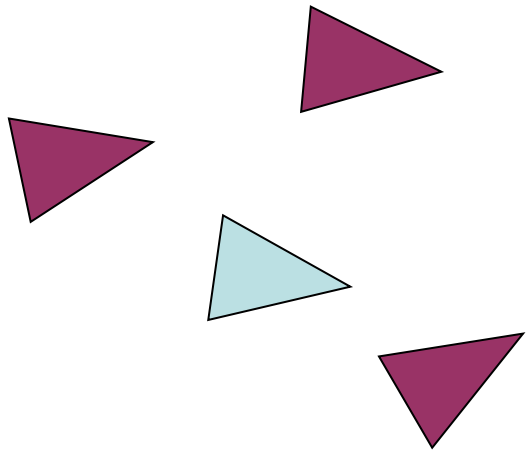


Alignment

Look at code!

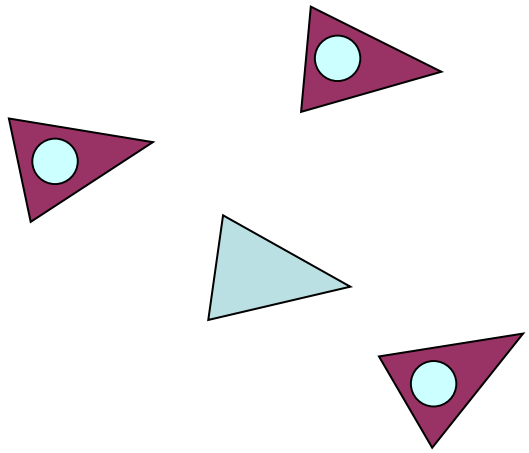
Cohesion

- Move agents towards neighbours’ centre of mass



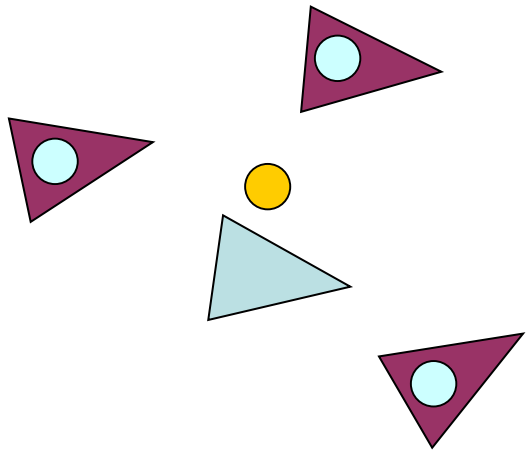
Cohesion

- Step 1: Find positions of neighbouring agents



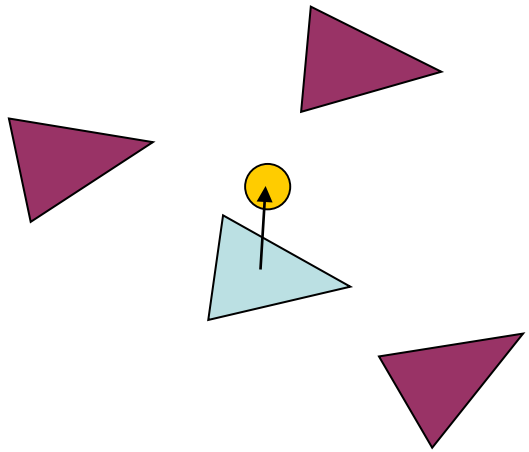
Cohesion

- Step 2: Compute average position



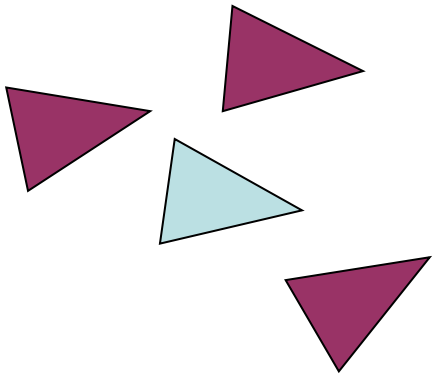
Cohesion

- Step 3: Seek to average position



Cohesion

- Result = agents stick together



Cohesion

Look at code!

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

Combining Group Behaviours

- Given 3 simple group behaviours for each agent:
  - Separation
  - Alignment
  - Cohesion
- We can create a wide variety of emergent collective behaviour
- Tweaking parameters of individual behaviours results in different group behaviour

Combining Group Behaviours

- Flocking
- Crowd path following
- Leader following
- Unaligned collision avoidance
- Queuing

http://www.red3d.com/cwr/steer/

Group Behaviours

- Audience

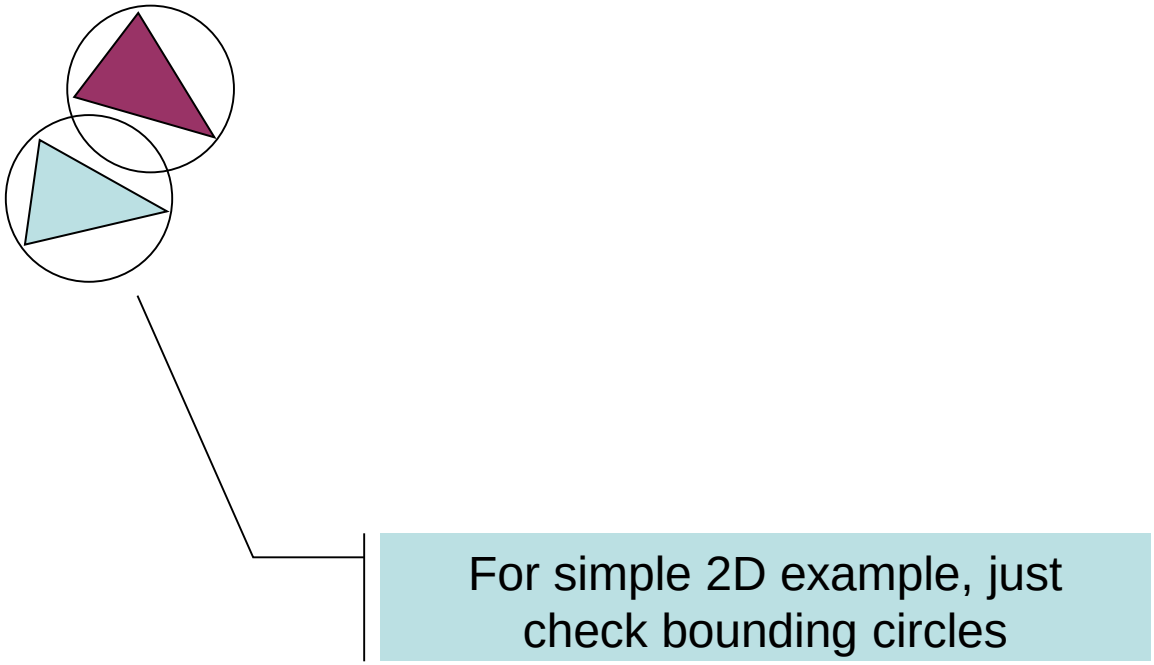
http://vimeo.com/1842245

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

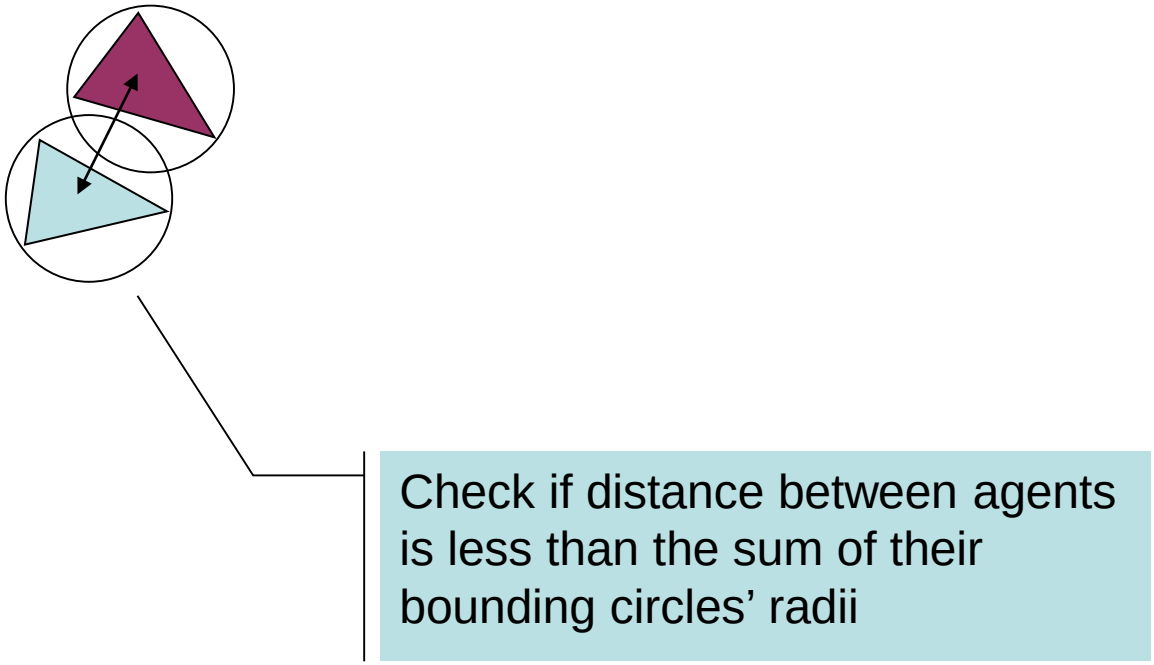
Zero Overlap

- Separation isn’t enough to guarantee agent non-penetration
- Add a non-penetration constraint



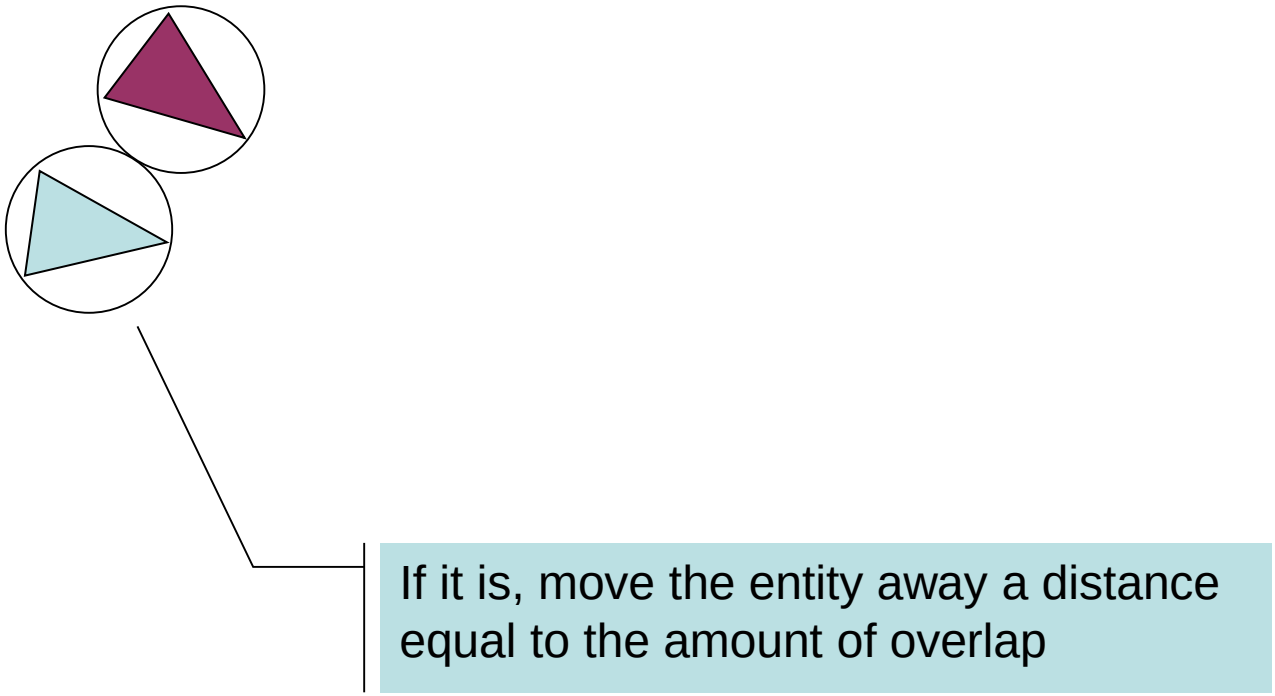
Zero Overlap

- Separation isn’t enough to guarantee agent non-penetration
- Add a non-penetration constraint



Zero Overlap

- Separation isn’t enough to guarantee agent non-penetration
- Add a non-penetration constraint



Zero Overlap

Look at code!

Zero Overlap

- Pros?
- Cons?

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview

Spatial Partitioning

- Many behaviours require testing against other agents
- Inefficient to test against every other agent
- E.g., for zero overlap:
  - If testing against 1 other agent takes 1 second
  - How much time for testing against 5 other agents?
  - How much time overall?

Spatial Partitioning

- Try to restrict the amount of testing needed
- One technique is spatial partitioning:
  - Partition the space up into cells
  - For a particular agent, determine which cell it is in
  - Only test other agents in that cell or in adjacent cells

- Other techniques:
  - BSP trees
  - Quad trees
  - Octrees
- Useful in many applications, not just A.I.

Spatial Partitioning

- Try a demo

Agenda

- Assignment 4 Redux
- Midterm
- Group Behaviours
- Combining Group Behaviours
- Zero Overlap
- Spatial Partitioning
- Assignment 5 Overview