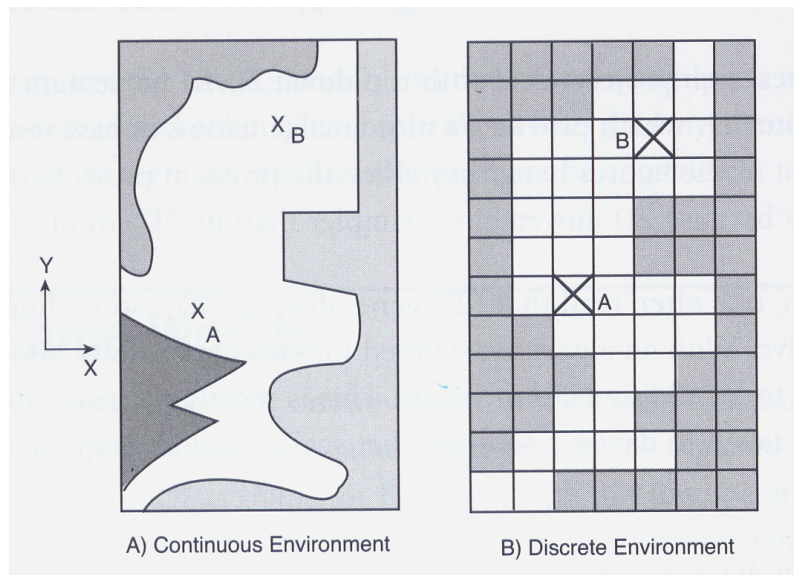# Navigation

- Navigation is the process of <u>purposefully steering</u> the course of an entity through a space.
- Navigation differs from plain movement
  - Plain movement could be due to such occurrences like an object falling off a cliff.

Navigation

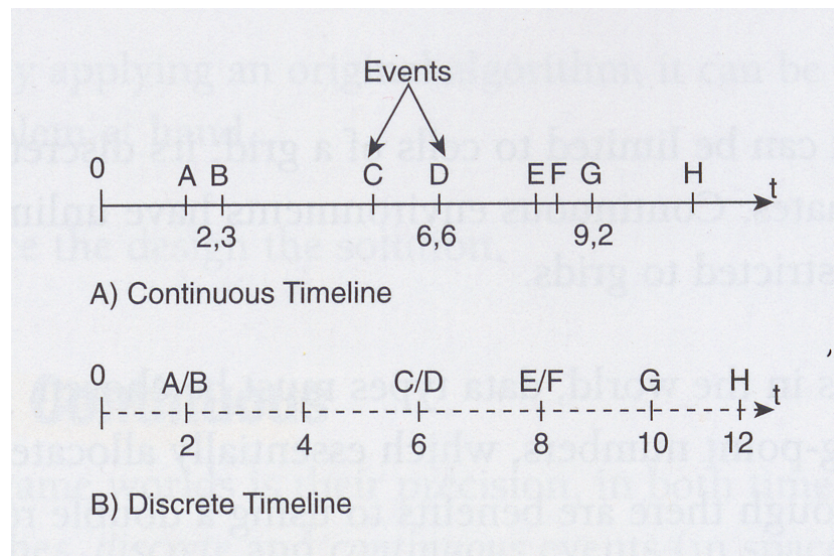- A game world describes a space.

Discrete
vs.
Continuous

A) Continuous Environment   B) Discrete Environment

☞ This has consequences on where items can be placed
and how animats can move from one position to the next.

Navigation

○ A game world describes time.

Discrete
vs.
Continuous

☞ This has consequences on how actions are perceived – smooth vs. choppy.
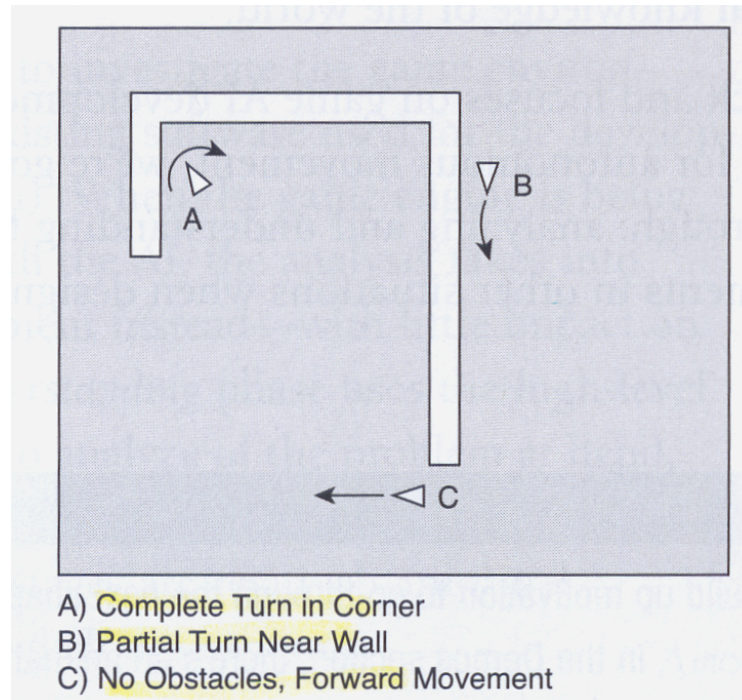
Navigation

- We want navigation to be
  - Realistic
    - avoid doing silly things
  - Efficient
    - it cannot be computationally expensive
  - Reliable
    - the same navigation strategies should work in many different scenarios
  - Purposeful
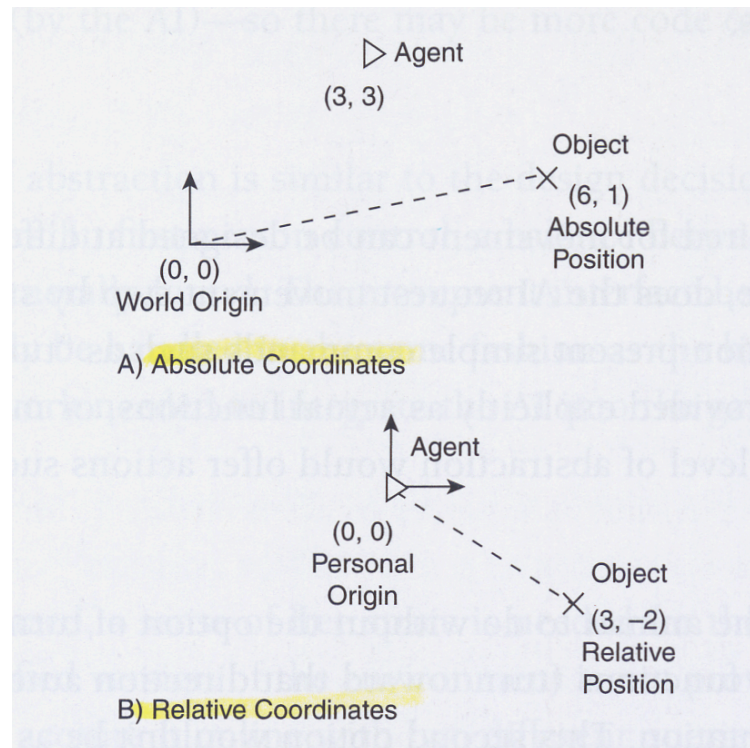   - it should serve some perceived goal

Navigation

- Example Scenarios – "Obstacle Avoidance Maneuvers"

A) Complete Turn in Corner
B) Partial Turn Near Wall
C) No Obstacles, Forward Movement

Navigation - Options
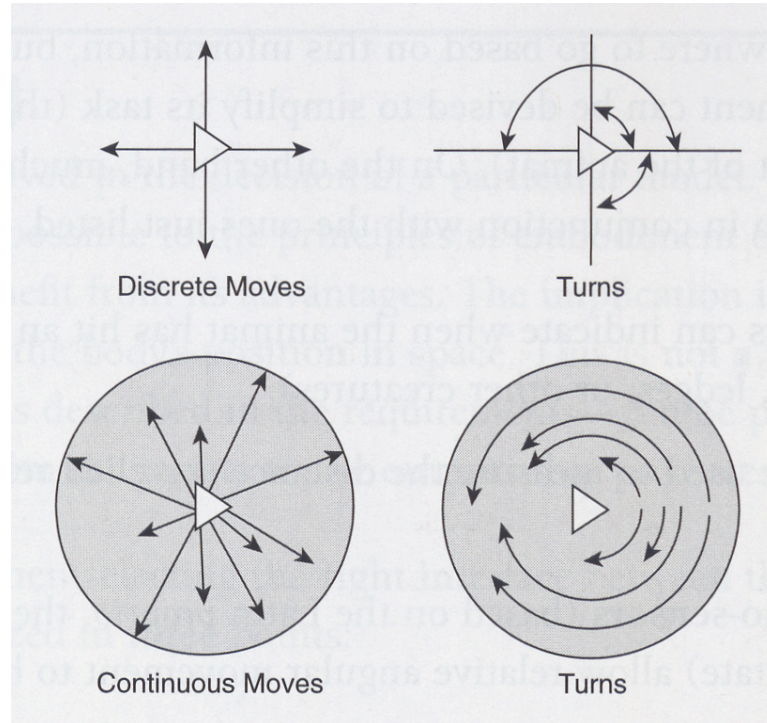
○ Animat Context

A) Absolute Coordinates

B) Relative Coordinates

☞ In the quagent API radius and rays calls return results in relative coordinates

☞ The where function returns results in absolute coordinates

Navigation - Options

○ Discrete vs. Continuous Actions

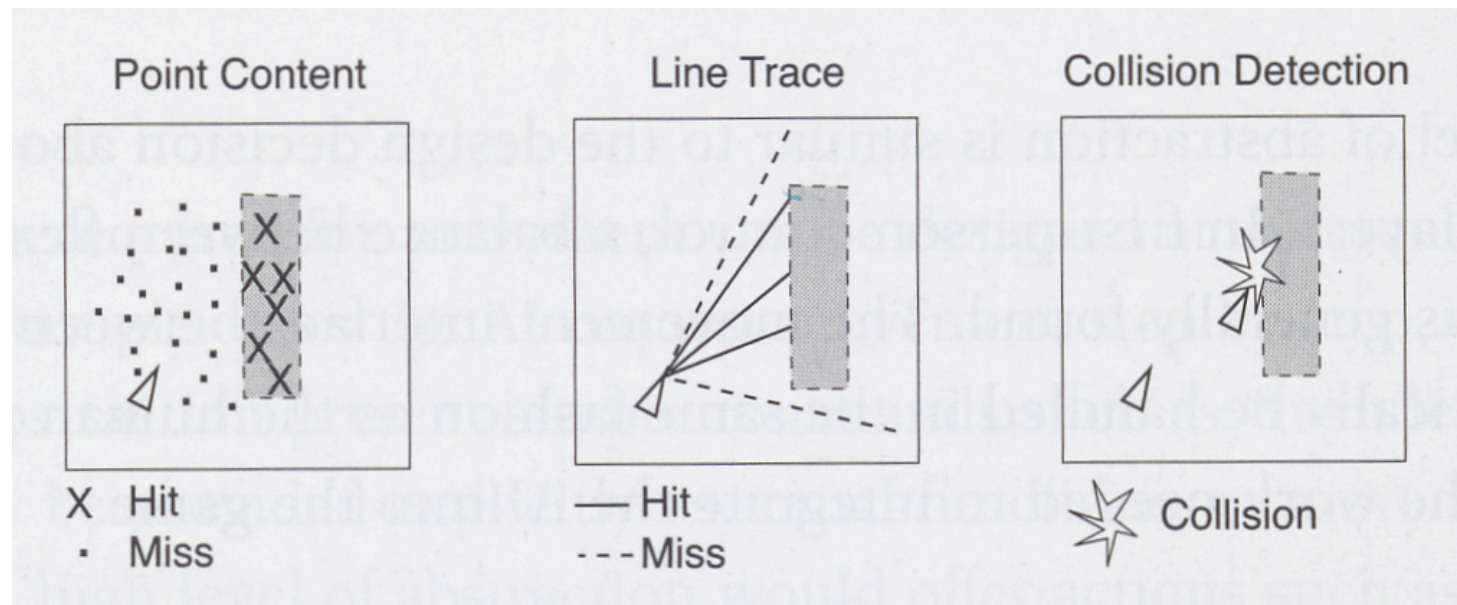☞ In the quagent world all actions are continuous

Navigation – Options

○ Senses

Point Content	Line Trace	Collision Detection
X Hit	— Hit	☆ Collision
· Miss	- - - Miss	

☞ Quagents implement point content with the radius command
☞ Quagents implement line trace with the rays command
☞Quagents implement collision detection with the TELL STOPPED event