Lecture 10/17:

Motion Planning for Finding Evasive Targets in a Cluttered Environment (also known as “How Robots Will Hunt You Down in Order to Use Your Biomass to Fuel their Robotic Empire”):

Whether or not it is antagonistic, target is hard to reach

Example: A point robot in an environment needs to find a target by sweeping the environment.

Target is unpredictable and can more arbitrarily fast

Environment is polygonal

Target/Robot are points

To “find” is to make a straight line between robot and target, unobstructed by obstacles

The robot's sub-region of sight must "block" the target from going into previously explored areas.

Does a solution always exist for a robot?

No, the robot can always have its vision blocked by an object from all angles, so the target can hide on the other side of the object, like two children running around a pillar. You may need >1 robot to catch the target

Effect of geometry on the number of robots

Tiny differences in geometry (orientation of niches in walls, etc) can change viability of single-robot solution.

How many robots?

Number n of edges: Depth of "Tree" determines number of robots – the number of “levels” of a tree

Number of robots = O(log(n)) [on the order of log(n)]

Number h of holes: The “worst case” arrangement of holes is in a square

Example: 25 holes (5x5), 5+2 robots (6 to go down grid, 1 to go across), Number of ‘bots is O(sqrt(h))

Total number of robots = O(sqrt(h)+log(n))

Information state

The visibility region is defined by free edges and obstacle edges

Robot records free edges (a, b, c) and whether they are cleared (0) or contaminated (1)

Sample Information State: (x, y, a=0, b=1, c=0)

Initial state: (x_0, y_0, a=1,b=1, ...u=1) where u is the last free edge

Goal state: (x_f, y_f, a=0,b=0, ...u=0)

Critical line:

Where the robot crosses that changes its information state (if it goes past the line it can change b=1 into b=0, etc), where a free edge becomes an obstacle edge.

If the robot stays within critical lines, it will not change the information state. This area is a “conservative cell”

Creating a tree

Nodes are different states of conservative cells and information states.

Recontamination: when the robot’s line of sight no longer blocks a free edge, the region is

contaminated again.

Certain maps can be solved, but only with recontaminating and reclearing

Two robots ('greedy' algorithm)

When the first robot cannot clear anymore, it stays in a fixed position while the second one clears more of the path ahead of it.

Robot with a cone of vision:

Stick to the wall to get the largest amount of vision possible.

Other topics

Dimensioned target (non-point) (how much of the target does the robot have to see?) and robots, 3d environments

Non-guaranteed strategies (probabilistic models)

Concurrent map building and target-finding

Planning the escape strategy of the target

10/2007 Ben Olson