Multi-Objective Path Planning in Virtual Environments

Tuğcem Oral

Advisor: Prof. Dr. Faruk Polat

May 4, 2011

Outline

Concepts on Problem

- Path planning
  - Offline / Online (Real Time) planning
- Objectivity
  - Single & Multi-objectivity
- Moving targets

Path Planning

- Finding a route from initial position to a target one.
- Online and offline methods.
- What if target moves?

Multi-Objectivity

- Thinking about optimization for more than one criteria.
    - Ex. Minimum path length (or associated time), minimum exposure to a hostile observer, safest path ...
- Domination between constraints.
- Pareto Optimal sets.

Moving Targets

- Could move intelligently / unintelligently.
- Pursuit / Catch operation.

The Problem Definition

- Finding paths toward target considering multiple criterias.
  - *Ex. Finding shortest and riskless paths.*
- Environment differs & matters!
  - Dynamic and partially observable environments are much more realistic.
- Existing solutions suffer from time complexity (fast planning).
- Come up with an algorithm covers all these aspects.

Studies centered on multiobjectivity

- Guo et. al [1] considers multi-objective path planning (MOPP) for the ball and plate system.
- Tarapata[2] presents selected multi-objective approaches to shortest path problems in his paper.
- Mitchell et al.[3] examine the problem through a low dimensional continuous state space subject to upper bounds on several additive cost metrics.

Studies centered on moving targets

- Goldenberg et al.[4] use multi-agent systems to pursue and capture the moving target.
- Hollinger et al.[5] concentrate on finding a moving, non-adversarial, known motion modelled target using multiple agents.
- Koenig and his team proposes an efficient incremental search algorithm in [6] and come up with D* lite in [7] for moving target search.

Other Alternative Methods

Generally evolutionary methods are used:

- Pangilinan et al. [8] introduce an evolutionary algorithm for multi-objective shortest path problem.
- Castillo et al. [10] define a genetic *offline* point-to-point agent path planner to find valid paths towards target.
- Nasrollahy et al.[9] proposes a particle swarm optimization algorithm as a multi-agent search technique for dynamic fully observable environments.

Outline

The Boundaries & Tradeoffs

- A Multi-objective real-time path planning algorithm should come up.
    - Objectives must be at least two, but also the algorithm should accept *n* objectives.
- Domination between objectives is crucial.
- Should run fast (due to real-time).
- Timing and synchronizing between concepts is important.

Outline

Environmental Properties

- 2-D grid environment.
- Partially observable and dynamic.
- For initial planning, target is stable on a pre-known location.
- Has an agent, threads and obstacles.

Obstacle Properties

- Might be both stationary and mobile.
- Some obstacles are transitive, but consumes time / energy.
  - *Ex. river, hill, etc...*
- Also intransitive obstacles.

Thread Properties

- Might be both stationary and mobile.
- Have *thread zones* where they have probability to shoot agent within.
  - The closer the agent, the more probable to shoot. - *shoot risk percent*
  - Zones can change by time.
- Do not try to catch or pursuit the agent.
- Mobile ones have movement schedule.

Agent Properties

- An *agent* enacts *path planner* role to navigate towards target.
- Has a sight area to observe environment.
- Only knows the location of target and withing its sight area at time t.
- Should re-plan its path every time step if world is updated.
- When enters to a thread zone, knows thread's movement schedule and move accordinly.
- Has life, when shot, it decreases.

What we have done so far

- Motivation is initialized.
- Related studies are examined.
- Problem definition is formalized.
- A basic structure for developing the algorithm is constructed.
    - Target is considered on a static location.
- Documentation of these studies are done.

Next Milestone

- Start to implement simulation environment.
  - While implementing, the predefined structures within algorithm will also develop.
  - Also, environmental concepts can be defined more precisely.

Further Studies

- We are at the very beginning of everything.
- Domain is not specified yet.
- Algorithm will be enhanced and test on simulated environment.
- Moving target concept should be added to environment.
- Formal documentation should be written.

Q & A

Any Questions?

- Feng Guo and Hongrui Wang and Yantao Tian, Multi-objective path planning for unrestricted mobile (2009), pp. 1046 -1051
- Tarapata, Zbigniew, Selected Multicriteria Shortest Path Problems: An Analysis of Complexity, Models and Adaptation of Standard Algorithms, Int. J. Appl. Math. Comput. Sci. (2007), pp. 269–287
- Mitchell, I.M. and Sastry, S., Continuous path planning with multiple constraints (2003), pp. 5502 - 5507 Vol.5
- Goldenberg, Mark and Kovarsky, Alexander and Wu, Xiaomeng and Schaeffer, Jonathan, Multiple agents moving target search, Morgan Kaufmann Publishers Inc. (2003), pp. 1536–1538
- Hollinger, Geoffrey and Singh, Sanjiv and Djugash, Joseph and Kehagias, Athanasios, Efficient Multi-robot Search for a Moving Target, Int. J. Rob. Res. (2009), pp. 201–219

Sun, Xiaoxun and Yeoh, William and Koenig, Sven, Efficient incremental search for moving target search, Morgan Kaufmann Publishers Inc. (2009), pp. 615–620

Sun, Xiaoxun and Yeoh, William and Koenig, Sven, Moving target D* Lite, International Foundation for Autonomous Agents and Multiagent Systems (2010), pp. 67–74

Pangilinan, J.M.A. and Janssens, G.K., Evolutionary algorithms for the multiobjective shortest path problem, World Academy of Science, Engineering and Technology

Amin Zargar Nasrollahy and Hamid Haj Seyyed Javadi, Using Particle Swarm Optimization for Robot Path Planning in Dynamic Environments with Moving Obstacles and Target, Computer Modeling and Simulation, UKSIM European Symposium on (2009), pp. 60-65

Castillo, Oscar and Trujillo, Leonardo and Melin, Patricia, Multiple Objective Genetic Algorithms for Path-planning Optimization in Autonomous Mobile Robots, Soft Comput. (2006), pp. 269–279