Multi-Target Adaptive A*

Kengo Matsuta Hayato Kobayashi Ayumi Shinohara

Graduate School of Information Sciences, Tohoku University, Japan

Introduction Agents often have to solve series of similar path planning problems. Adaptive A* [Koenig and Likhachev 05] is a recent incremental heuristic search algorithm that solves such problems faster than A*, updating a heuristic function using information from previous searches. We address path planning with multiple targets on Adaptive A* framework. We consider two cases whose objectives are (1) an agent has to reach one of the targets (**OR setting**) , and (2) an agent has to reach all of the targets (**AND setting**).

OR setting : An agent must find a shortest path reaching the nearest target.

MinPlan

Initial formula

$$h_{or}(v) := \min_{\gamma \in \Gamma} h_{\gamma}(v)$$

Γ : a set of targets

h_{γ} : ordinal heuristic function for target γ

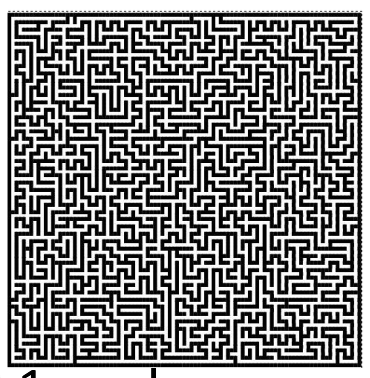
Update formula for moving target

$$h_{or}^{t+1}(v) = \max\{\min_{\gamma \in \Gamma^{t+1}} H^t(v,\gamma),\quad h_{or}^t(v) - \max_{\gamma \in \Gamma^{t+1}} h_{or}^t(\gamma)\}$$

Theorem

h_{or} and h_{or}^{t+1} are consistent.

Experiment



- Maze : four-neighbor mazes of size 100 × 100.
- Initial positions of the agent and targets are chosen randomly.
- Agent does not know terrain of the maze initially.
- At each step,
  - each target moves to an adjacent unblocked cell with probability 0.1, and
  - at most one cell is selected with probability 0.1 and its status is changed (blocked / unblocked).

Results (average of 500 trials)

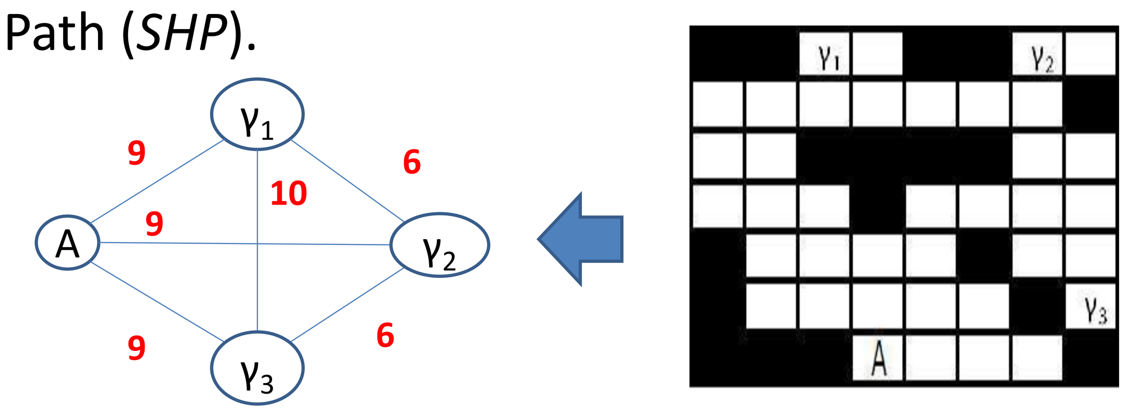
	#targets	#expansions	runtime[ms]
NaivePlan (Calculate the shortest paths to all goal cells at each search by Adaptive A*)	5	2880903	1195
	10	6092105	2652
	15	8377218	3589
MinPlan	5	18097	23
	10	8428	13
	15	5520	9

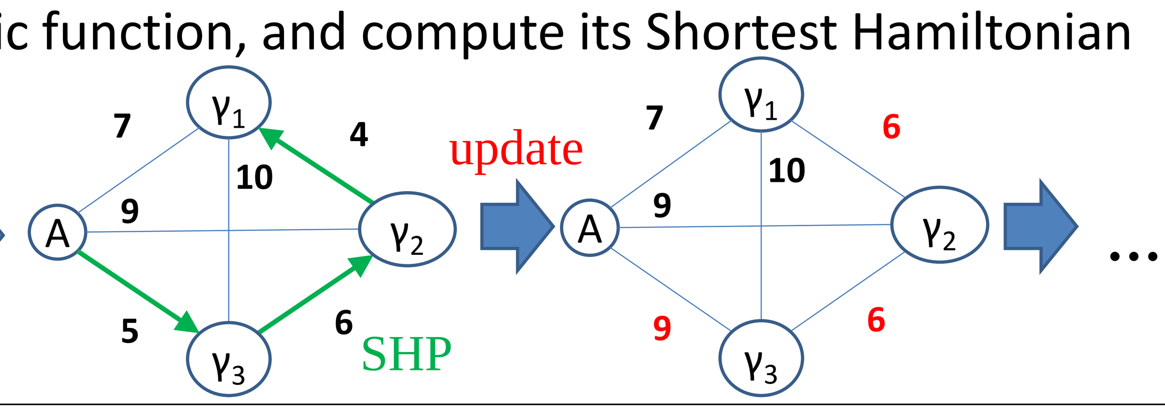
- Runtime of MinPlan gets lower when the number of targets increases, because target tends to position near by agent.

AND setting : An agent must find a shortest path reaching all of the targets.

Abstract Graph Approach

Consider an abstract graph G which represents (approximate) distances among the agent and all targets based on the heuristic function, and compute its Shortest Hamiltonian Path (SHP).





(l : #edges, m : #cells, n : #targets)

StraightforwardPlan $O(n^2(l \log m + 2^n))$

1. Compute *all* distances by Adaptive A*.
2. Calculate SHP p^*_{shp} in G and return it.

Theorem

$$C(p^*_{and}) = C'(p^*_{shp})$$

IncrementalPlan $O(n^2(l \log m + n^2 \cdot 2^n))$

1. Calculate SHP p^*_{shp} in current G.
2. If all distances of the edges in p^*_{shp} that are already computed in Step 3, then return p^*_{shp}.
3. Compute *only* the (uncomputed) distances of the edges in SHP by Adaptive A*, and update the heuristic function and G, and go back to Step 1.

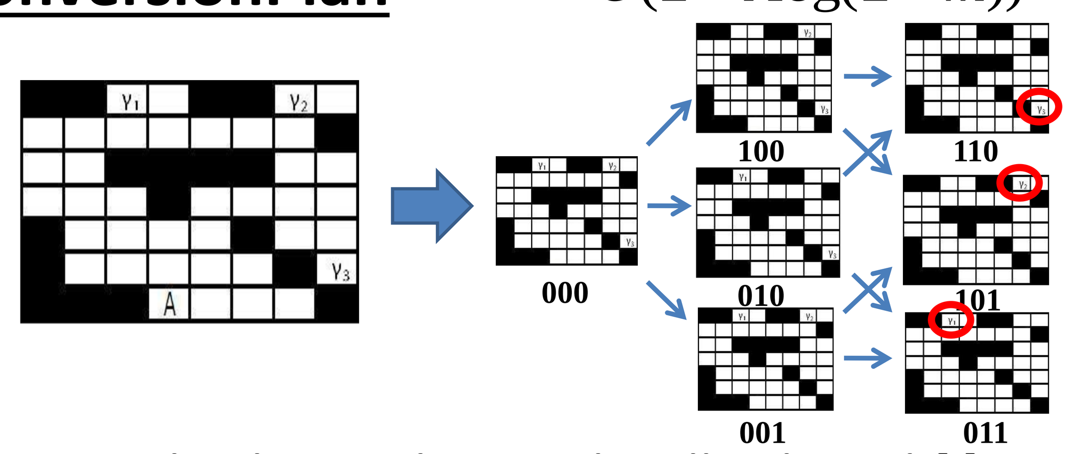
Theorem

$$C(p) = \min_{p' \in P_{inc}(V)} C'(p')$$

Conversion Approach

Convert to OR setting.

ConversionPlan $O(2^n \cdot l \log(2^n \cdot m))$



1. Attach a binary b to each cell, where $b[i]$ represents whether an agent has already visited a goal cell γ_i .
2. Solve the converted problem in OR setting, whose goals are γ_i with 1^n .

Theorem

$$C(p^*_{and}) = C'(p^*_{or})$$

Experiment

- Maze : four-neighbor mazes of size 100 × 100.
- Initial positions of the agent and targets are chosen randomly.
- Agent knows terrain of the maze initially.
- No target moves, and all cells are fixed through a trial.

Results (average of 100 trials)

	StraightforwardPlan		IncrementalPlan		ConversionPlan	
#targets	#expansions	runtime[ms]	#expansions	runtime[ms]	#expansions	runtime[ms]
4	15936	3	10455	2	27227	9
6	32133	6	19340	4	111743	74
8	54660	11	28963	8	501397	435
10	89479	19	40889	23	1707189	2091

- ConversionPlan is quite slower than the other methods in the same tendency of its worst case time complexity.
- IncrementalPlan is faster than StraightforwardPlan when $n \leq 8$.

$\Rightarrow$ IncrementalPlan is practically more efficient than StraightforwardPlan when n is relatively small.

Conclusion We formalized OR and AND settings. For OR setting, we proposed a construction method of a consistent heuristic function to utilize Adaptive A*. For AND setting, we proposed three methods to directly utilize Adaptive A* for each target. We also proved that all of the methods always achieve an optimal path of AND setting. Our experimental results showed that the proposed methods properly work on an application, i.e., maze problems, both in OR and in AND settings.

The 9th International Conference on Autonomous Agents and Multiagent Systems, May 12, 2010, Toronto, Canada.