AI: A* on Terrain Maps

Kevin Wang, Philip Chen

February 15, 2016

1 Heuristics

1.1 Exponential Cost Function

Exponential Cost Function

1.2 Divisive Cost Function

Division Cost Function

For the second cost function, a simpler heuristic is used. Here is the cost function:

$$Cost(h_1, h_2) = \frac{h_2}{h_1 + 1}$$

Different than the other cost function, the value of the cost function is based on the height values, not just the difference in heights. Here is the heuristic:

Heuristic

C = Chebychev distance
N = Node height
E = Endpoint height
m = min(N, E)

$$h(n) = \frac{m}{m+1} * C$$

The idea is to relax the "rule" that one can only travel above land. This heuristic calculates the cost if one walked in a straight line, with each step the same height as the previous, until the end point.

Listing 1: Heuristic Code for Divisive Function

```java
private double getHeuristic(final TerrainMap map, final Point pt1, final Point pt2)
{
    double z1 = map.getTile(pt1);
    double z2 = map.getTile(pt2);
    double dist = Math.max(Math.abs(pt1.x-pt2.x), Math.abs(pt1.y-pt2.y));
    double min = Math.min(z1, z2);
    return dist*min/(min+1);
}
```

Consistency and Admissibility

A heuristic is consistent if and only if the following is true:

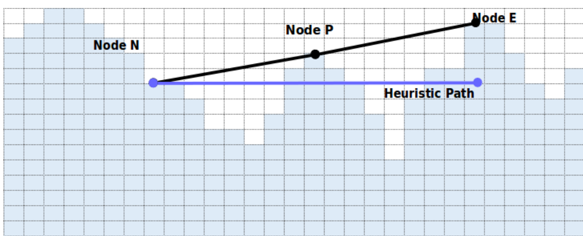
$$h(N) \leq Cost(N, P) + h(P)$$

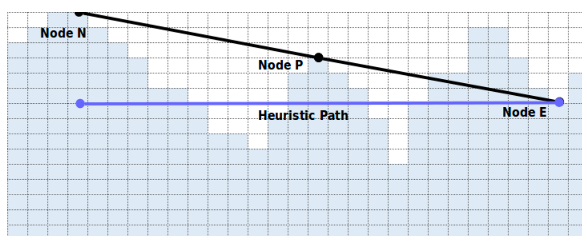
P is a neighbor of N, and in the explanations, can even be expanded to any point between N and the end-point. There are four cases for this, where n, p, and e are the heights of nodes N, P, and E respectively: $n < p < e$, $p < n < e$, $e < p < n$, $e < n < p$.

n <p <e and e <p <n:

In these cases, the height used by the heuristic is lower than or equal to all other heights in the path. By the cost AND the heuristic equation, this flat, low height path will also be lower than the actual path cost.

In the examples below, the heuristics use path costs clearly lower than the actual costs. They also satisfies the consistency equation $h(N) \leq Cost(N, P) + h(P)$.



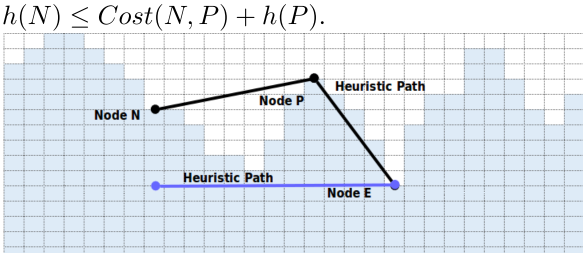


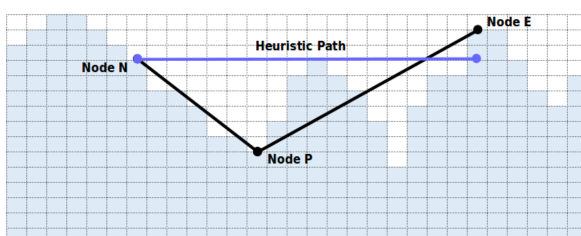
In these examples, the blue shaded cells are not the actual cost, but just to help with visualization. The actual path is outlined by the bolded lines, N being the current node, E being the end point, and P being an intermediary node.

e <n <p and p <n <e:

In these more likely cases, the path to the end is not monotonic, rather it contains a change in direction, or mini hills or valleys. In this case, it may seem that because there is an extra downhill segment, the path could be potentially decreased. But all of the extra uphill that needs to be made up causes other segments to be more steep, thus having a greater cost.

In the examples below, although the paths dip below the heuristic's heights, the added steepness to the other segments more than make up for the slightly lowered costs. In fact, traveling along hills and valleys are more costly than traveling along a consistently sloped path. Although it is not mathematically proven for all cases in this report, the examples, and many like it, satisfies the consistency equation $h(N) \leq Cost(N, P) + h(P)$.





In these examples, the blue shaded cells are not the actual cost, but just to help with visualization. The actual path is outlined by the bolded lines, N being the current node, E being the end point, and P being an intermediary node.

2 Heuristic Efficiency

2.1 AStarExp

	Seed 1	Seed 2	Seed 3	Seed 4	Seed 5
Path Cost	533.4482191461119	549.5036346739352	510.97825243663607	560.6570436319696	479.5879215923168
Uncovered	71644	81810	74001	66382	67837
Time Taken	1104	1213	1070	1060	996

2.2 AStarDiv

	Seed 1	Seed 2	Seed 3	Seed 4	Seed 5
Path Cost	198.59165501141644	198.56141550095256	198.4864317411443	198.70066424826052	198.25499446810
Uncovered	1013	1004	1005	1006	1020
Time Taken	58	58	54	52	58