Implementing Anya

Daniel Harabor

March 18, 2014

This paper describes some technical details for implementing Anya: an online and optimal pathfinding algorithm. For a description of the algorithm and its properties refer to [1].

1 Preliminaries

We work with grids consisting of W×H square cells. Each cell is an open set of points that are all either *traversable* or *non-traversable*. Edges in the grid are horizontal or veritcal transitions between the corners of square cells. We model edges as open intervals of *intermediate points*. There are several other types of points which are interesting. These are shown in Figure **??**.

In this work we want to compute paths which are sequences of points of the type $\langle p_1, \ldots p_k \rangle$. In particular we want each path to have the property that every point p_i is *visible* from both p_{i-1} and p_{i+1}. Examples involving pairs of points that are both visible and non-visible are shown in Figure **??**.

For further details refer to [1].

2 Overview

Anya can be thought of as a particular implementation of A* search. It uses the same principles as that algorithm: there is an open and a closed list, there is an admissible heuristic function (in this case, Euclidean distance) and the search proceeds by generating and expanding nodes in the order of minimum f-value. The only thing that differentiates Anya from other algorithms is (i) how we define search nodes and (ii) how we identify the successors of each search node.

3 Search Nodes

A node is a tuple (I, r) where I is a contiguous interval and r is a root point. Every point $p \in I$ is visible from r. :w

References

[1] Daniel Damir Harabor and Alban Grastien. An optimal any-angle pathfinding algorithm. In *ICAPS*, 2013.