Requirements Analysis

SP Bosbrand Specifications Team sp-bosbrand@cs.uu.nl

April 26, 2003

1 Introduction

This specification is split up in four main thematic parts. In each of the sections we present the list of expectations of the project provider and also our proposed solutions to those problems. Technical details and aspects of the implementation were not considered.

2 World

The world has certain global attributes.

- windpower
- winddirection
- humidity

These attributes are the same for all cells that constitute part of the forest ('world').

2.1 Static items

Each static item has the following properties.

fuel determines duration cell is on fire.

threshold below which it burns

state not burning, burning, burnt

cost

The following static items will be implemented in the "final" version.

Trees Tree, we only keep track of one kind of tree that has the following attributes.

Grass Grass.

Water Our primary interest is that it doesn't burn.

Road Promenade routes in the forest. These can be large enough so that a bulldozer can drive on it, reducing the time of the travel. Unlike a fire-line, fire *can* cross this cell.

Fire-line We will use the word 'fire-line' to describe the part of the forest that has been cleaned by the bulldozers. Fire cannot cross this cell. [1]

Building This is expensive when burnt down so the controller should avoid this (represents residential area, industrial zone, etc.).

Depot The start point of the ground fire-fighting agents, deposit place for bulldozers. They are expensive when burnt.

2.2 Dynamic cells

The coordinates and attributes of these cells are determined at runtime.

Fire Represents a burning cell.

Waterbomb Region that decreases the fire activity.

Ground Agent Represents a bulldozer, which is able to create a fireline, trough grass or trees. This agents starts in a depot.

Air Agent Flying agent, drops waterbombs on the fire.

The behaviour of the agents is determined by the controller.

3 Simulation

The simulation will be modeled using Cellular Automata (4 neighbours, no cyclic boundary conditions). It determines the fireactivity for tree's and grass using the cell's attribute(s). The propagation of the fire will be stochastic.

4 Controller

The basic task for the controller is to direct all active agents.

[1] This may be misleading because the same word is sometimes used to refer to the concept of to the meeting point between the fire and the yet untouched forest, these being the points where the fire has the highest activity. We will refer to this last concept as 'fire-front'.

It calculates subtargets for each agent and generates the most optimal route for the agents. Evalution of subtargets is implemented using neural networks (NN) of which the weights are 'trained' using ESP.

The shortest route between the subgoals is determined using dynamic path planning algoritmes (implemeted using "A*" or "Dijkstra's kortste pad algoritme").

4.1 General demands

- Agents begin at selected location(s).
- The location of (sub)targets are dynamic and wil be relocated.
- When multiple agents are active, they will cooperate.
- An initial startpoint is chosen according to the four or eight winddirections (N,O,Z,W).
- The fire-line is an closed polygon, the last subtarget coincides with the first.
- The location of subtargets will be constantly re-evaluated.
- There will be only one fireline, even when using multiple agents.

4.2 Agents

There will be two kinds of agents, airborne units and ground units.

4.2.1 Ground Agents demands

- Agents will not enter the fire.
- Subtargets are unique per agent.
- Agents are able to dig a fire-line, through grass or trees. Trees require more time than grass.
- When possible the agents will use water as part of the fire-line.
- The fireline will be created as precisely as possible according to the planned route.
- Units move faster on existing roads.

4.2.2 Air Agents demands

- Agents will drop waterbombs on strategic locations in the fire.
- Agents have to refuel water at an airport or in a lake.
- A waterbomb will only slow a fire down.

4.3 Path Planning

The path planner will plan the path from start location to first subtarget. From there on it will plan from subtarget to subtarget. The path will be optimized to minimalize costs. The demands are listed in ascending order, specific to the costs of the actions.

- Riding on existing roads
- Riding on gras
- Riding trough trees
- Digging of grass
- Digging trough trees

5 Userinterface

The application clearly has two modes of operation, *learning* the agent's NN using ESP and *visualizing* a single run after the NN are trained. Therefor we will provide two seperate interfaces.

5.1 Learning Mode

This interface provides statistical information on the population etc. It also allows modification of the ESP parameters and related parameters such as those of the random world generator.

It can be run autonomously and will store the (possibly intermediate) NN and other results of the process for later examination.

5.2 Visualization Mode

The userinterface provides a convenient world editor for building customary worlds.

It furthermore allows visualization and modification of parameters related to the simulation (wind speed, direction, start point of fire, humidity).

This allows modifying (and saving/storing) configurations. These can be used for training new NN (using the "learning mode") or for demonstrating existing NN on.