1 Problem Description

Given an infinite grid graph with vertex set $\mathbb{Z}^2$ and edges $\{((u,v),(x,y)) \mid |u-x|+|v-y|=1\}$ i.e. each cell is neighbor to the cell directly above, below, left, and right. W.l.o.g. a fire starts at $(0,0)$ and spreads over time. After each time step, all cells with a burning neighbor start burning as well. To stop the fire from spreading, at each time step we can place a certain number f of agents on non-burning vertices to protect them. These vertices can then never catch fire. f can either be an integer to tell exactly how many vertices can be protected at each step, or f tells how many vertices can be protected on average.

The program uses an evolutionary algorithm to solve two different problems.

Enclosing Fire The objective is to totally stop the fire from spreading. Formally this means that there has to be some point in time where no burning vertex has a non-burning, unprotected neighbor.

Highway-Protection The objective is to protect a highway from catching fire as long as possible. A highway H is a horizontal line of vertices at a certain distance m to the origin $H := \{(i,m) \mid i \in \mathbb{Z}\}$.

2 Model

To solve these problems the algorithm uses two approaches to model different strategies.

- A *scattered strategy* consists of a sequence of coordinates which tells the order of protected vertices. In principle this allows to model every possible strategy.
- A *connected strategy* focuses on protecting vertices such that these form a single connected barrier.

3 Program Usage

3.1 Setup

The class 'EvoHighwayProtection.java' contains the main function. Parameters can be set in the 'parameters.config' file. A line starting with a # is considered a comment. Each parameter is explained within the file.

3.2 While running

While the program is running, it will print an ASCII-art preview of the best strategy to the console so you are able to judge its progress. It will do so at most once every second. While it is running, you can type the following commands

show prints an ASCII-art preview of the best strategy to the console

steps prints the development of the best strategy in ASCII-art to the console

stop stops the simulation and saves the results

reset restarts the simulation

3.3 View Results

The state of a grid is saved in text files ending with “.grid”. A grid file has one line for each $y-$coordinate and each line has a value for each $x-$coordinate. Numbers encode the status of a cell

100	empty cell
-100	protected cell
0-9	burning cell the number is depending on the distance to the outbreak point

You can use the provides files “plot_finalGrid.gp”, “plot_history.gp” and “plot_steps.gp” to plot the results using gnuplot. The history has one plot of the best strategy in the population whenever it improved. Using the ’steps’-plot you can see how the finally best barrier is built during the spread of the fire.

Furthermore, the fitness is saved in the file “fitness.log” which can be plotted using “fitnessDevelopment.gp”.