

MARS COLONIZATION PROGRAM

Shreya Laddha, Neha Singhal, Mukta Wagle

Rover Path Finder

Introduction	2
Features	2
Shortest Path with Multiple Destinations	
Rover Race	
Dynamic Node Size	
Dynamic Statistics	
Grid Size	
Instructions Panel	
Other modifications	
Summary	4

Introduction

The aim of the project is to modify and optimize the existing PathFinding.js library as a part of Mars Colonization Program, Microsoft Engage 2020. The project is a browser based web application to help the Mars Curiosity Rover find the shortest path to the destination while avoiding obstacles on the way. It also includes finding the most optimal path covering multiple destinations on the way and a fun feature of race between rovers. Some of the other key features are dynamic node size and dynamic statistics.

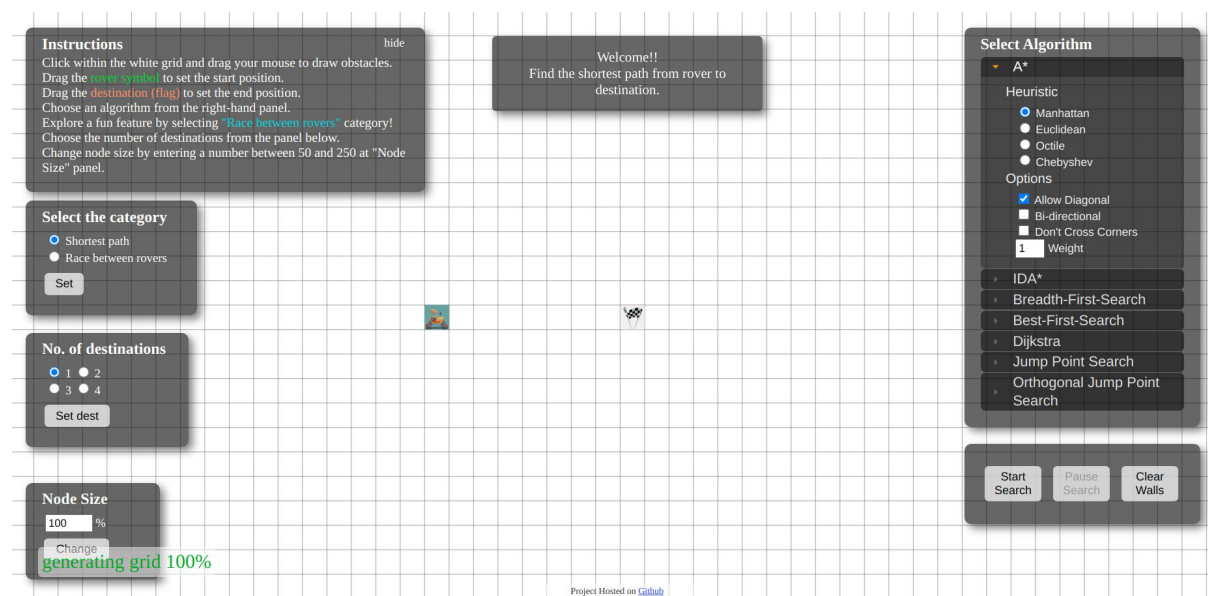
Features

The project has a lot of new features which makes it more fun for the user to use. The modified user interface makes the experience of the user better in many ways. All the additions and modifications are as follows -

SHORTEST PATH WITH MULTIPLE DESTINATIONS

The homepage of the web application opens with a welcome message on top which disappears after a few seconds. The default category is shortest path and default number of destinations is one. The rover starting position is displayed by a rover image on the grid and the destination is a flag image on the grid. The user can drag the rover and the flag as well as draw obstacles. After selecting a suitable algorithm, the user can start the search. Once the path is found, it is displayed on the window with a cadet-blue line and the corresponding statistics are shown in bottom-left corner. If there is no path such that the rover can reach the destination, a popup is displayed alerting the user of the situation.

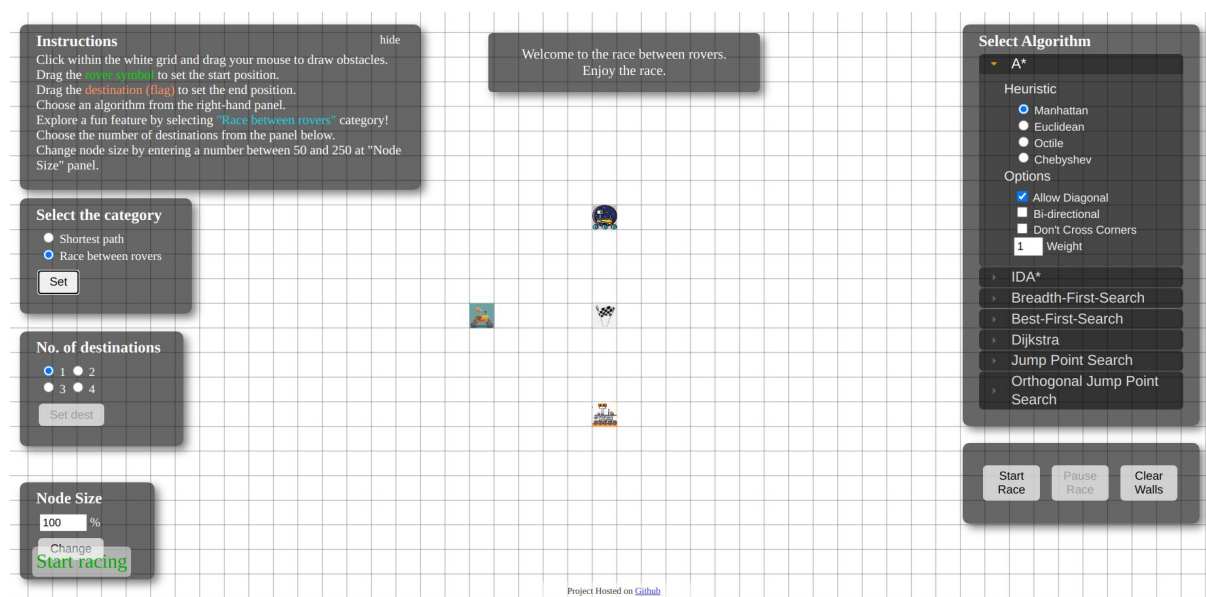
This simple path search feature is extended and applied to finding the most optimal path covering multiple destinations. The user can select the desired number of destinations from No. of destinations panel in the left side of the window, after which the corresponding number of Flag images are displayed in the grid. On searching, if any of those destination is not reachable, path not found popup appears in the window.



ROVER RACE

The most interesting feature of the project is the rover race. The user can switch to this feature by choosing 'Race between Rovers' option in 'Select the Category' panel. The number of destinations button is disabled in this category and is re-enabled on changing back to 'Shortest Path' category. A welcome message appears on the top of the page displaying - 'Welcome to the race between rovers. Enjoy the race'.

Three different rovers are located in a grid around one destination point. User can drag the rovers and the destination in the grid as well as introduce obstacles. Once we finalise the algorithm and start the search, the shortest path is found for each rover. Then each rover starts taking one step towards the destination simultaneously, and the steps taken can be seen by the user which gives the illusion of an actual race. The statistics of the length of the path of each rover is displayed in the bottom-left corner of the window. A popup message informs the user about the winner or winners, if there is a tie. In case a rover cannot reach the destination in any way, the same is displayed in the popup message as well as statistics.



DYNAMIC NODE SIZE

This feature helps the user to change the node size of the grid simply by entering a desired number in the text box in Node Size panel. On clicking the change button, a new grid is generated without any change in the category selected in the Category panel. The user must enter a number between 50 to 250 in the text box or else a popup appears instructing the user to do so. Thus, this feature allows a user to increase or decrease the node size and hence the number of cells in the grid according to user requirement.

DYNAMIC STATISTICS

Dynamic statistics are displayed in the bottom-left corner of the window, describing every state that the application is in. When the category is changed, it displays - Start searching or Start racing accordingly. Every button on the search algorithm panel is associated with its own state, depending on the category, which is shown in statistics so the user can always keep a track of what is going on and in which state the application is. For example - on clicking the pause search/race button, 'Paused. Start searching/racing again' is displayed in the statistics. When the path search is completed, the statistics display the length, time and operations of the search. In case when no path is found, 'Path Does not Exist' is shown in place of length. For the rover race, the length of the path of each of the rover is displayed in the statistics so that user can always know by how much distance a rover lost.

GRID SIZE

The application has a feature of dynamic grid size. The grid size is adjusted to the current window's width and height, so that no cell of the grid lies in an area invisible to the user. This way the path that can exist will always be visible to the user. In case of maximise/minimize the window, the user can reload the page to get the grid again adjusted to the visible window.

INSTRUCTIONS PANEL

Since, a lot of space is taken by this panel on the window, there is a hide option on the panel. On clicking hide, the instructions panel is hidden and the other three panels on the right side move up from their place and an additional line 'See instructions' is displayed in the footer. The latter feature comes in handy when the user wants to see the instructions later on. On clicking the link in the footer, the instructions panel is again displayed but this time in the top-center of the window as opposed to the default top-right corner. The user can always hide the panel.

OTHER MODIFICATIONS

In source code of the algorithms, the bi-algorithms are combined into mono-algorithms, and the algorithms are optimized. The JumpPointFinder has only two diagonal movements, hence some unused and redundant files have been deleted. Benchmark and tests folders were added. The user can test the algorithms by 'npm test' command. Benchmark file can be accessed by 'node benchmark.js' command in benchmark folder. Trace search is similar to BestFirst search, hence not added in the application.

Summary

The project incorporates many different features so as to elevate user experience and extend the use of the application for fun purposes as well. The aim of the project was to help navigate the rover to the destination in shortest path. But in addition to this, the application can also serve as a game in which the crew members can randomly select any place for their rover in the grid with obstacles and then race among them. Quite a fun activity and good luck with your choice of position !! Moreover, the features can be easily extended to incorporate more number of rovers and destinations, implying no restrictions on the capability of the application.

