

**Autonomous Systems**

Lab #2 - Potential Functions

**Delivered by:**

Mazen Elgabalawy u1999109

Solomon Chibuzo Nwafor u1999124

**Supervisor:**

Alaaeddine El Masri El Chaarani

**Date of Submission:**

06/11/2024

**Table of Contents**

[1. Final Outcome: 3](#_Toc181896665)

[1.1. Grid-Map: 3](#_Toc181896666)

[1.2. Attraction Potential: 4](#_Toc181896667)

[1.3. Distance Maps 5](#_Toc181896668)

[1.4. Repulsive Potential 6](#_Toc181896669)

[1.5. Total Potential 7](#_Toc181896670)

[1.6. Gradient Descent 8](#_Toc181896671)

[1.7. Wave-Front Planner 9](#_Toc181896672)

[2. Implementations: 13](#_Toc181896673)

[2.1. Attraction Potential Function: 13](#_Toc181896674)

[2.2. Brushfire Algorithm: 14](#_Toc181896675)

[2.3. Repulsive Potential Function: 15](#_Toc181896676)

[2.4. Total Potential Function: 15](#_Toc181896677)

[2.5. Gradient Descent Algorithm: 15](#_Toc181896678)

[2.6. Wavefront Planner Algorithm: 16](#_Toc181896679)

[2.7. Find Path: 16](#_Toc181896680)

[3. Challenges: 18](#_Toc181896681)

[3.1. Testing Brushfire and Wavefront planner: 18](#_Toc181896682)

[4. Conclusion: 18](#_Toc181896683)

# Final Outcome:

The following sub-sections show the results for each exercise of the lab for different maps.

## Grid-Map:

Figures 1 shows the grid-maps used for testing the different functions in this Lab. All the grid-maps are made by importing their respective images and then setting black cells (occupied) to a value of 1 and white cells (free) to a value of 0. Table 1 shows the start and goal points for each map that will be used in the next sections.

A purple and yellow square with white text

Description automatically generatedA yellow and purple square with a number of lines

Description automatically generated with medium confidence

A maze with a chart

Description automatically generated with medium confidenceA yellow and purple square with numbers and lines

Description automatically generated

Fig.1: Grid-maps for different maps that were used to test our implementations

|  |  |  |
| --- | --- | --- |
| Map | Start | Goal |
| 0 | (10,10) | (110,40) |
| 1 | (60,60) | (90,60) |
| 2 | (8,31) | (139,38) |
| 3 | (50,90) | (375,375) |

Table.1: Start and Goal points for each map

## Attraction Potential:

Figure 2 shows the Attractive potential for each map, calculated using the quadratic potential energy from the goal point of each map provided in Table 1. The value of was set equal to 0.001 in all maps.

A chart with numbers and a square

Description automatically generated with medium confidenceA chart with numbers and a square

Description automatically generated with medium confidence

A chart with numbers and a square

Description automatically generated with medium confidence A chart with numbers and a square

Description automatically generated with medium confidence

Fig.2: Attractive Potential for each map for goal points presented in Table.1

## Distance Maps

Figure 3 shows the distance maps calculated using the **Brushfire** algorithm, to calculate the distance between a cell and its nearest obstacle.

A map with a gradient of blue and green colors

Description automatically generated with medium confidenceA map with a gradient of colors

Description automatically generated with medium confidence

A maze with a maze in the middle

Description automatically generated A map with numbers and a gradient of colors

Description automatically generated with medium confidence

Fig.3: Distance Maps for each map calculated using Brushfire algorithm

## Repulsive Potential

Figure 4 shows the repulsive potential field for all the maps calculated the distance maps from section 1.3. The value of (repulsive force strength) is 5 and Q (threshold distance beyond which potential is zero) is 30.

A yellow and purple square with numbers

Description automatically generatedA yellow and purple square with white text

Description automatically generated

A maze with a chart

Description automatically generated A chart with yellow shapes on a purple background

Description automatically generated

Fig.4: Repulsive Potential Field for every map calculated with and

## Total Potential

The Total Potential Field of the Grid-map is calculated as the sum of the Attraction and Repulsion fields. Figure 5 shows the total potential fields for all the maps.

A chart of a field

Description automatically generatedA chart with a number of lines

Description automatically generated with medium confidence

A maze with a maze in the middle

Description automatically generatedA chart with numbers and a gradient

Description automatically generated with medium confidence

Fig.5: Total Potential Field for all maps

## Gradient Descent

This section shows the results of applying the **Gradient Descent** algorithm to the total potential field from section 1.5. Gradient Descent was applied using both connectivity 4 and connectivity 8. Figure 6 shows the gradient descent using connectivity 4, while Figure 5 shows the algorithm with connectivity 8. Gradient descent gets stuck in local minimum in most cases, thus the need for a different planner.

A chart of a gradient

Description automatically generated with medium confidence A yellow and purple square with a red star

Description automatically generated

A maze with a red dot in the middle

Description automatically generated A yellow and purple square with a red star

Description automatically generated

Fig.6: Gradient Descent applied with connectivity 4 to all maps.

A yellow and purple square with a red star

Description automatically generated A yellow and purple square with a red star

Description automatically generated

A maze with a red dot in the middle

Description automatically generated A yellow and purple chart with numbers and a red star

Description automatically generated

Fig.7: Gradient Descent applied with connectivity 8 to all maps.

## Wave-Front Planner

This section shows the results for both the **Wave-Front Planner** and the Path generated using the resulting wave-map. The wave map was made using both connectivity 4 and 8, and those maps were used to find the path using both types of connectivity. Figures 8 and 9 show the wave-maps and the path generated using connectivity 4 respectively, while figures 10 and 11 show the wave-maps and paths generated using connectivity 8 respectively. The **Wave-Front Planner** can find a path in all maps and does not get stuck at local minimums.

A diagram of a wave map

Description automatically generatedA diagram of a diagram

Description automatically generated

A maze with a wave map

Description automatically generatedA map with different colors

Description automatically generated with medium confidence

Fig.8: Wave-map using connectivity 4

A diagram of a path using wavefront using connectivity

Description automatically generatedA diagram of a diagram

Description automatically generated

A maze with a red dot in the center

Description automatically generatedA yellow and purple chart with a red star

Description automatically generated

Fig.9: Path generated using Wave-Front Planner using connectivity 4

A chart with a gradient of purple and green colors

Description automatically generated with medium confidenceA diagram of a diagram

Description automatically generated

A maze with a maze and a maze with a maze and a maze with a maze and a maze with a maze and a maze with a maze and a maze with a maze and a maze with

Description automatically generatedA diagram of a wave map

Description automatically generated

Fig.10: Wave-map using connectivity 8

A diagram of a path using wavefront using connectivity

Description automatically generatedA diagram of a diagram

Description automatically generated

A maze with a path to the center

Description automatically generatedA graph with yellow shapes

Description automatically generated

Fig.11: Path generated using Wave-Front Planner using connectivity 8

# Implementations:

This section includes the implementation flow charts for all the algorithms used in this lab.

## Attraction Potential Function:

Figure 12 shows the Flow diagram for the **Attraction Potential Function.**

**A white and black background with a diamond

Description automatically generated**

Fig.12: Flow Chart for Attraction Potential Function

## Brushfire Algorithm:

Figure 13 shows the Flow chart for implementing the **Brushfire** algorithm.

A screenshot of a black screen

Description automatically generated

Fig.13: Flow Chart for Brushfire.

## Repulsive Potential Function:

Figure 14 shows the flow chart for implementing the Repulsive Potential Function

A screenshot of a computer

Description automatically generated

Fig.14: Flow Chart for Repulsive Potential Function

## Total Potential Function:

The total Potential Function is the sum of the Attractive Potential Function and the Repulsive Potential Function.

## Gradient Descent Algorithm:

To make sure that the algorithm does not get stuck in an infinite loop, we used a threshold of 500 for the number of iterations that can be done before returning the path. Figure 15 shows the flow diagram for the algorithm.

A screenshot of a cell phone

Description automatically generated

Fig.15: Flow Chart for Gradient Descent.

## Wavefront Planner Algorithm:

We used a similar method to the **Brushfire** algorithm in section2.2, but the wave expands from the goal instead of the obstacles. The flow chart for **Wavefront Planner** is shown in figure 16.

## Find Path:

Finally, using the generated wave-map from section 2.6, we apply a version gradient descent that follows the minimum of the neighbors until it reaches the goal. Figure 16 shows the flow chart for **Find Path**.

A screenshot of a cell phone

Description automatically generated

Fig.16: Flow Chart for Wavefront planner.

A screenshot of a cell phone

Description automatically generated

Fig.17: Flow Chart for Find Path function.

# Challenges:

## Testing Brushfire and Wavefront planner:

Because of the relatively large size of the grid-maps, testing the **Brushfire** algorithm while implementing was difficult, as there was no easy way to see where the fault was happening (due to the large number of iterations, visualizing the map at each iteration or debugging the code is almost impossible).

The solution to this was to make a small 10x10 grid-map as a testing environment. This allowed for easy visualization of the evolution of the distance-map, making debugging the algorithm easier and more rapid. The same map was used to test the **Wavefront-Planner**. The grid-map mentioned is shown below.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **0** | **0** |
| **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |
| **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **0** |
| **0** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **0** |
| **0** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **1** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |

# Conclusion:

In this lab we explored the use of potential functions as a method for path planning. We used the gradient descent algorithm to create a path from a start configuration to a goal configuration, however, the algorithm usually gets stuck at local minimum. Hence, it is recommended to use the Wave-front planner as it does not have a local minimum and can always find a path from start to goal configurations.

It is also recommended to normalize the attraction and repulsive potential functions in order to match their scales to prevent one of the maps being more dominant than the other in the total potential function, ensuring that the gradient descent can work across different environments.