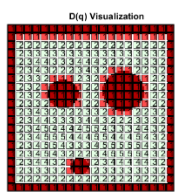
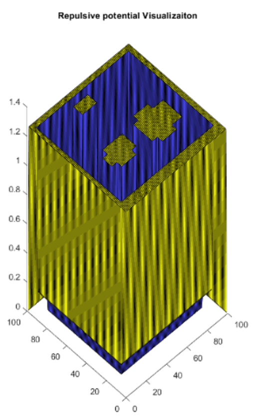
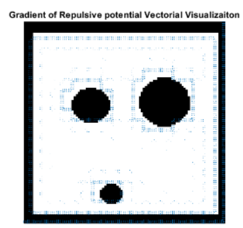
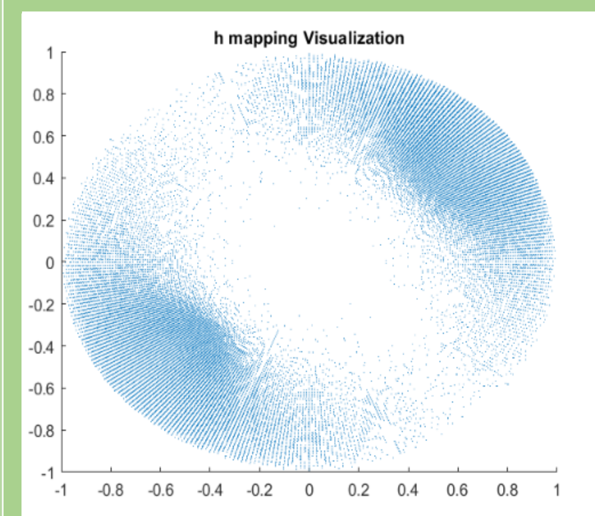
ROBOT MOTION PLANNING (ME510)

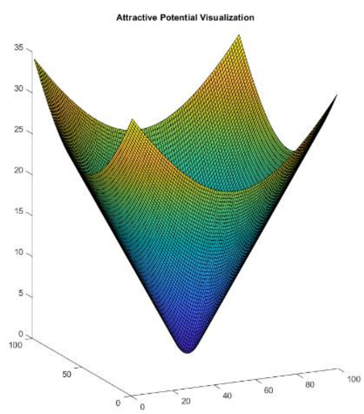
Assignment 2

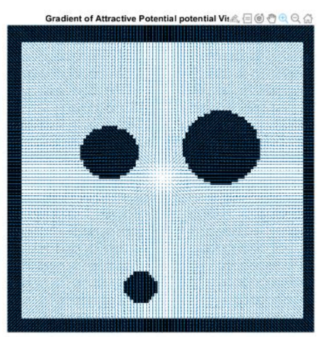


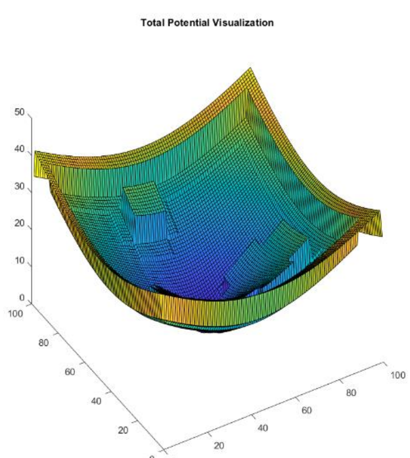


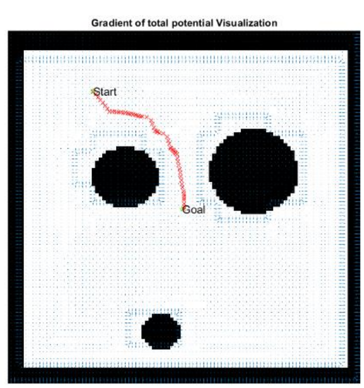












ADITYA SHAH (Roll:2011mt02)

Department of Mechatronics

For the Academic Year 2021 - 2022

Path Planning Using Potential Functions

- Robot is assumed as a positive charge
- Goal as negative charge
- Obstacle as positive charge
- Positively charged robot will get attracted towards negatively charged goal
- But Positively charged robot gets repelled by positively charged obstacle
- Robot follows a path by following negated gradient of potential function
- We know, Conic Potential poses discontinuity problem at the origin causing chattering and Quadratic Potential grows without bound as q moves away from q_goal.
- To overcome the limitation of both Quadratic and Conic Potential, I have used both the potential to get more desired Attractive Potential Function.
- So, Conic potential attracts the robot when it is very distant from q_goal and the Quadratic potential attracts the robot when it is near $qgoal$,ths 8vercoming each other limitation.

$$U_{\text{att}}(q) = \begin{cases} \frac{1}{2}\zeta d^2(q, q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ d^*_{\text{goal}}\zeta d(q, q_{\text{goal}}) - \frac{1}{2}\zeta (d^*_{\text{goal}})^2, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}. \end{cases}$$

$$\nabla U_{\text{att}}(q) = \begin{cases} \zeta(q - q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ \frac{d^*_{\text{goal}}\zeta(q - q_{\text{goal}})}{d(q, q_{\text{goal}})}, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}, \end{cases}$$

Where, d^*_{goal} is the threshold from the goal where the planner switches between conic and quadratic potential.

- Repulsive Potential repel the robot when robot comes under domain of influence of Obstacles ,else no repulsion.

$$U_{\text{rep}}(q) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{D(q)} - \frac{1}{Q^*}\right)^2, & D(q) \le Q^*, \\ 0, & D(q) > Q^*, \end{cases}$$

$$\nabla U_{\text{rep}}(q) = \begin{cases} \eta \left(\frac{1}{Q^*} - \frac{1}{D(q)}\right) \frac{1}{D^2(q)} \nabla D(q), & D(q) \le Q^*, \\ 0, & D(q) > Q^*, \end{cases}$$

Where, Q^* is the region of influence of obstacles for the robot.

- Finally using Gradient Descent technique, Robot finds a path to goal and at goal we are using epsilon , considering tolerance of 0.1, because $\nabla U(qi)$ reduces but does not vanish at goal during implementation.

Description of Brushfire:

Algorithm Descriptions

- The brushfire algorithm relies upon a grid-based world, either 4 or 8 connected.
- Obstacles are initialized to a value of 1, while the free space is initialized to zero.
- The grid is iteratively searched, when a pixel is found with a value of i, its adjacent zero-valued pixels are set to i+1.
- This process continues until no more zero-pixels are found in the graph.
- The value in the cell represents the distance to the closest obstacle and this is used in Repulsive Potential for D(q) .

Equation:

$$U_{\text{att}}(q) = \begin{cases} \frac{1}{2}\zeta d^2(q, q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ d^*_{\text{goal}}\zeta d(q, q_{\text{goal}}) - \frac{1}{2}\zeta (d^*_{\text{goal}})^2, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}. \end{cases}$$

$$\nabla U_{\text{att}}(q) = \begin{cases} \zeta(q - q_{\text{goal}}), & d(q, q_{\text{goal}}) \leq d^*_{\text{goal}}, \\ \frac{d^*_{\text{goal}}\zeta(q - q_{\text{goal}})}{d(q, q_{\text{goal}})}, & d(q, q_{\text{goal}}) > d^*_{\text{goal}}. \end{cases}$$

$$U_{\text{rep}}(q) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{D(q)} - \frac{1}{Q^*}\right)^2, & D(q) \leq Q^*, \\ 0, & D(q) > Q^*, \end{cases}$$

$$\nabla U_{\text{rep}}(q) = \begin{cases} \eta \left(\frac{1}{Q^*} - \frac{1}{D(q)}\right) \frac{1}{D^2(q)} \nabla D(q), & D(q) \leq Q^*, \\ 0, & D(q) > Q^*, \end{cases}$$

Parameter Description:

Eta: gain on the Repulsive gradient, **set to 4.5**

Zeta: gain on the attractive force, **set to 0.1**
DStar: threshold distance from the goal where the planner switches between conic and quadratic potentials, **set to 5**

Nu: Weightage to Repulsive Potential, **set to 6**

Dq: Value **read from the brushfire graph**

QStar: Minimum distance to consider the obstacle, **set to 4**

Alpha: Gradient Descent Parameter, **set to 2.1**

Epsilon: Tolerance for Grad at goal position, **set to 0.1**

<u>Attractive Repulsive potential technique is not complete</u>

- **Problem of Local Minima**

Illustrations:

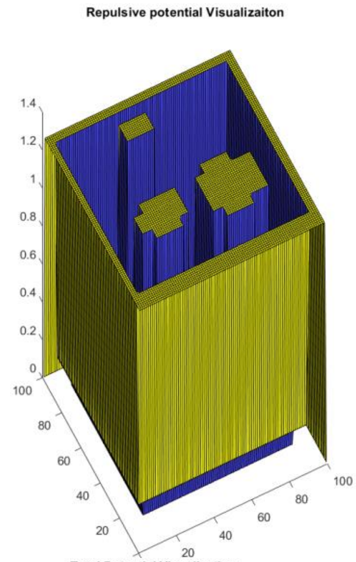
For following Configuration:

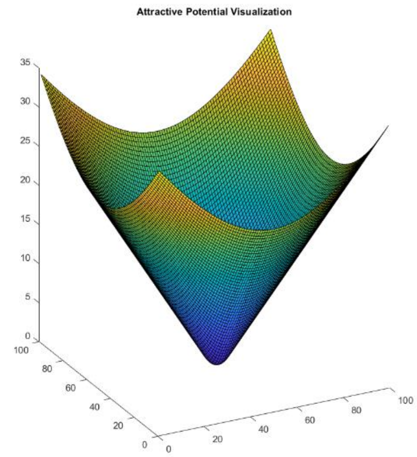
Command Window

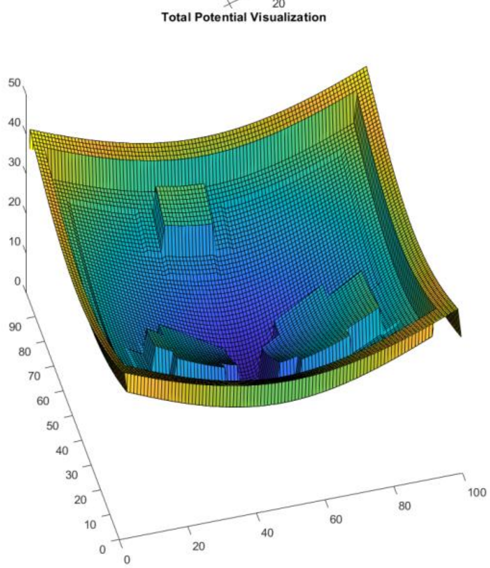
```
Type in a value of x goal: 50
Type in a value of y goal: 51
Type in a value of x start: 25
Type in a value of y start: 18
fx
```

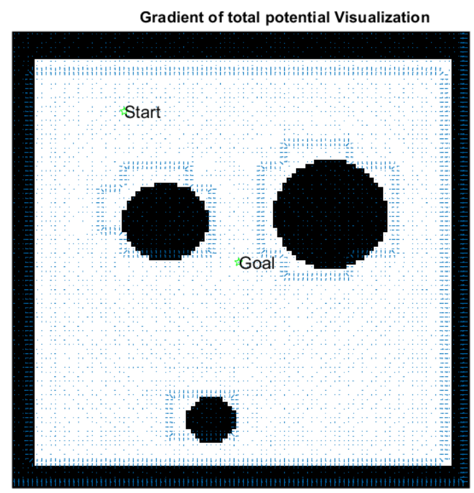
1st Case:

- Eta:4.5
- Zeta:0.1
- DStar:5
- Nu:6
- QStar:4
- Alpha:2.1
- Epsilon:0.1





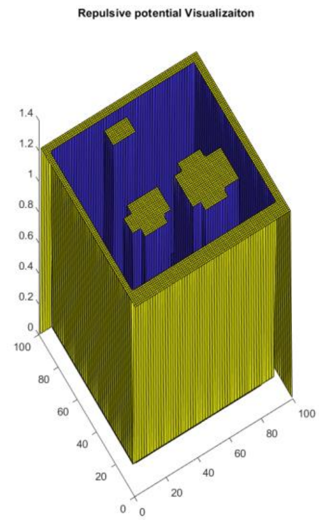


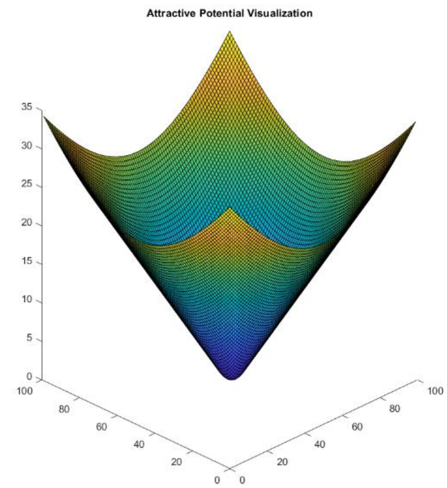


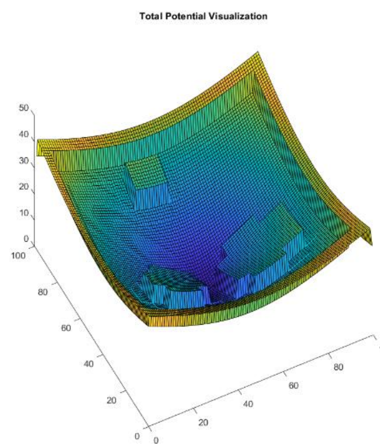
Analysis: We can see that near the obstacles, we are having high repulsive potential and its influence decreases gradually as distance to obstacles increases. And at Goal position the Potential is minimum and smooth.

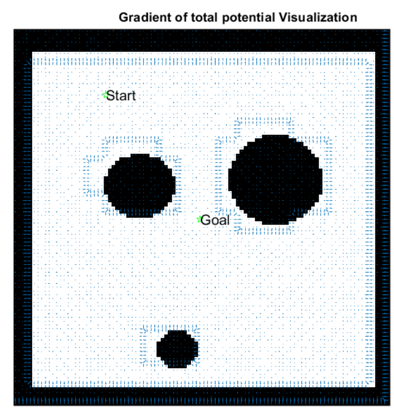
2nd Case:

- Eta:5.5
- Zeta:0.1
- DStar:5
- Nu:6
- QStar:3
- Alpha:2.1
- Epsilon:0.1





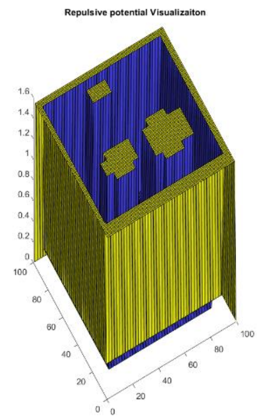


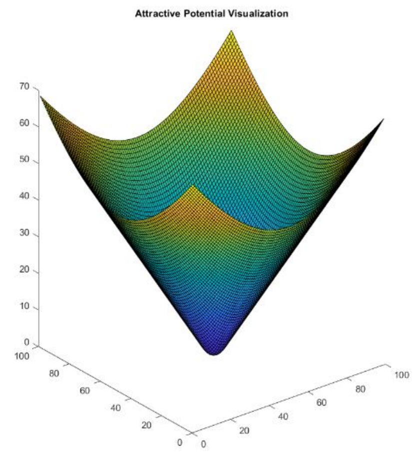


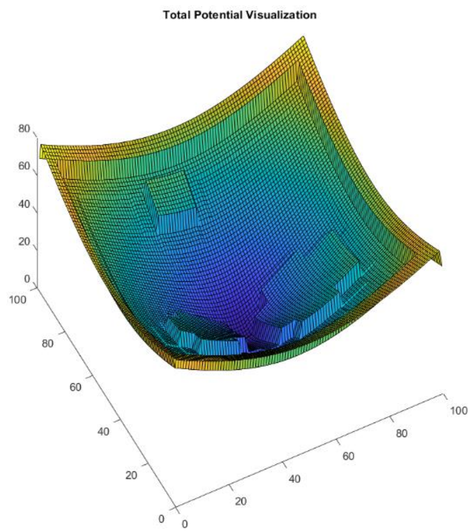
Analysis: Here we will observe because of lower QStar and higher Eta, Repulsive Potential has more stronger influence but the region of influence has reduced,w.r.t to previous one.

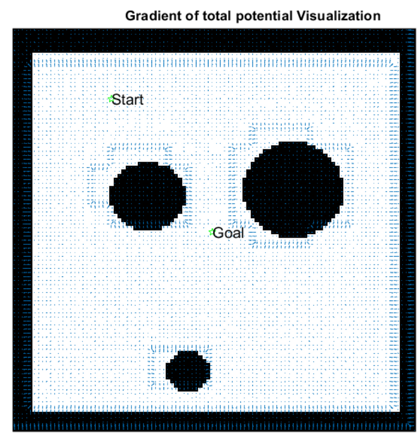
3rd Case:

- Eta:5.5
- Zeta:0.2
- DStar:5
- Nu:6
- QStar:4
- Alpha:2.1
- Epsilon:0.1





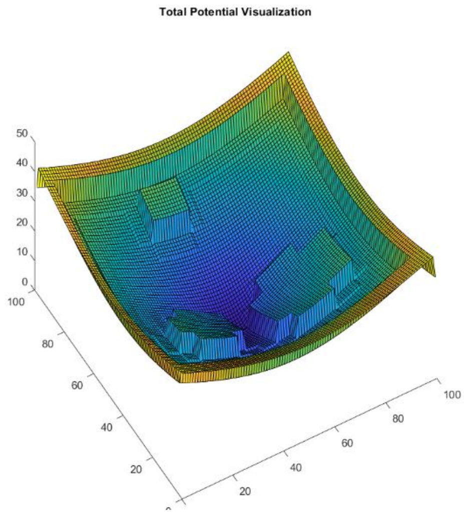


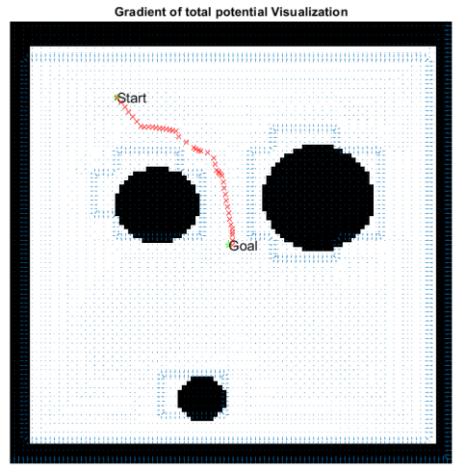


Analysis: Here we observe that the attractive potential effect has increased as Zeta has increased w.r.t previous case .

4th Case:

- Eta:4.5
- Zeta:0.1
- DStar:5
- Nu:6
- QStar:4
- Alpha:2.6
- Epsilon:0.1





Analysis: Here we observe w.r.t case 1, the iteration time has decreased significantly as the alpha value is increased little bit. Thus, convergence rate has increased but the it has to be tuned in such a manner such that it doesn't lead to oscillation due to several overshoot near goal location.

Decision for Proper Tuning taken During Final Implementation:

- Eta: gain on the Repulsive gradient, **set to 4.5**

Given an weightage to Repulsive potential and for conservative approach given a , higher value >1 is selected and by analysis ,I found 4.5 to be good as it gave good results .

- Zeta: gain on the attractive force, **set to 0.1**

Given an weightage to Attractive Potential, but < 1 as high attractive potential may lead to instability and also to get a smoother gradient(used both quad and conic potential function). So, given 0.1 after analysis.

- DStar: threshold distance from the goal where the planner switches between conic and quadratic potentials, **set to 5**

Choosing the key point for shift of Attractive potential from Conic to Quadratic Attractive Potential,considering the distance from q to q_goal.

- Nu: Weightage to Repulsive Potential, **set to 6**

Conservative Approach→ environment with some more weightage to Repulsive Potential given and after analysis found as 6.

- QStar: Minimum distance to consider the obstacle, **set to 4**

Chosen from brushfire algorithms, for considering the region of influence of obstacles and thus chosen 4 .

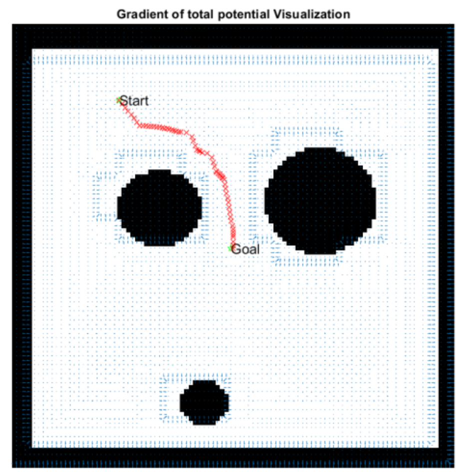
- Alpha: Gradient Descent Parameter, **set to 2.1**

Grad Descent Parameter ,chosen in such a manner so no overshoot /oscillation occur and iteration rate is also good. So chosen a value >1 ,giving good results with no overshoot. Found to be 2.1.

- Epsilon: Tolerance for Grad at goal position, **set to 0.1**

Considering the tolerance considerable ,for the iteration to end for gradient descent .

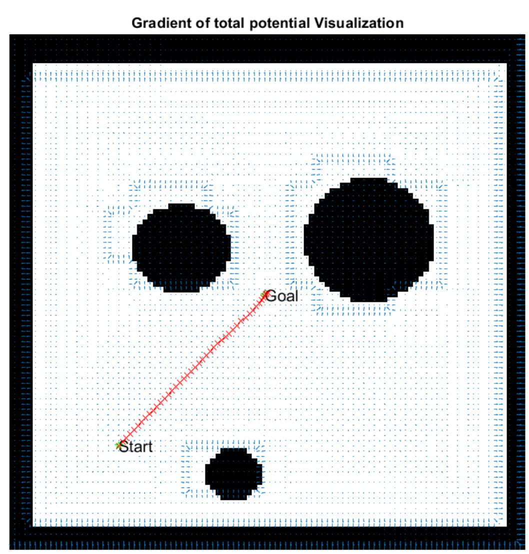
Testing Completeness With Given Parameter:



- Eta:4.5
- Zeta:0.1
- DStar:5
- Nu:8
- QStar:4
- Alpha:2.1
- Epsilon:0.1

- Case 1:
- Q_goal=[50,51];
- QStartPosition=[25, 18];

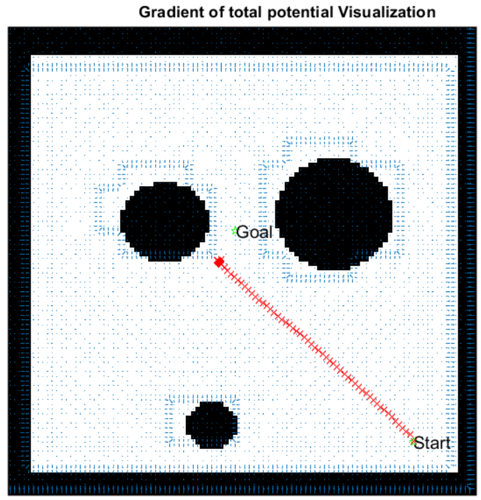
➢ **Reached Goal**



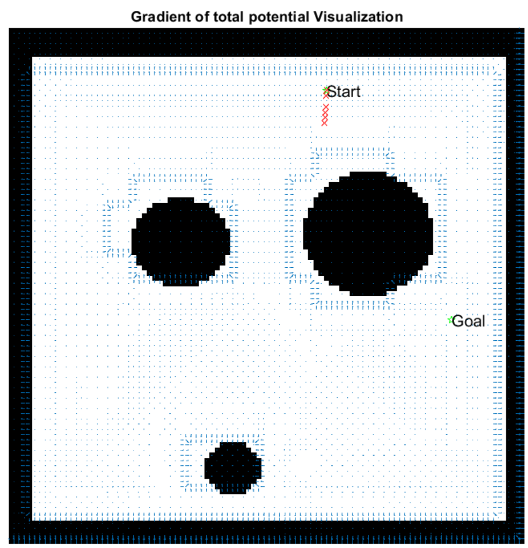
- Case 2:
- Q_goal=[50,51];
- QStartPosition=[22,80];

➢ **Reached Goal**

- Case 3:
- Q_goal=[49,44];
- QStartPosition=[87,89];
  - **Goal not reached.**



- Case4:
- Q_goal=[86,57];
- QStartPosition=[62,13];
  - **Goal not reached.**



<u>Inferences:</u>

Thus, from above we can say that the given solution for Path Planning is **not complete.**

- **Attractive Repulsive potential technique is not complete.**

It led to local minima instead of reaching Goal. Also, after proper tuning for parameter, it can have local minima problem in the space.

3rd Question: **Navigation Function Approach with Star Domain Mapping.**

Equation used:

- h mapping between the star- and sphere-spaces constructed using a translated scaling map

$$T_i\,(q) = v_i\,(q)(q - q_i\,) + p_i, \text{where } v_i(q) = \left(1 + \beta_i(q)\right)^{0.5} \frac{r_i}{d(q,q_i)}$$

Where, qi is the centre of the star-shaped set,

pi and ri are, respectively, the centre and radius of the spherical obstacle

and βi (q) defines a star-shaped set

- For star-shaped obstacle OQi, analytical switch is defined as

$$\mathrm{s_i}(\mathrm{q}, \lambda) = \left(\sigma_\lambda o \frac{\gamma_\kappa \bar{\beta}_i}{\beta_i}\right)(q) = \frac{\gamma_\kappa \bar{\beta}_i}{\gamma_\kappa \bar{\beta}_i + \lambda \beta_i} \bar{\beta}_i(q) = \prod_{j=0, j\neq i}^{n} \beta_j$$

- For goal the analytical switch is defined as:

$$s_{q_{goal}}(q, \lambda) = 1 - \sum_{i=0}^{M} s_i$$

- The mapping is thus defined as:

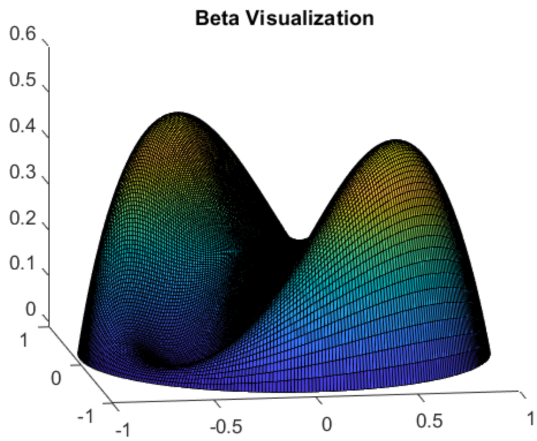
$$h_\lambda(q) = s_{q_{goal}}(q, \lambda) T_{q_{goal}}(q) + \sum_{i=0}^{M} s_i(q, \lambda) T_i(q)$$

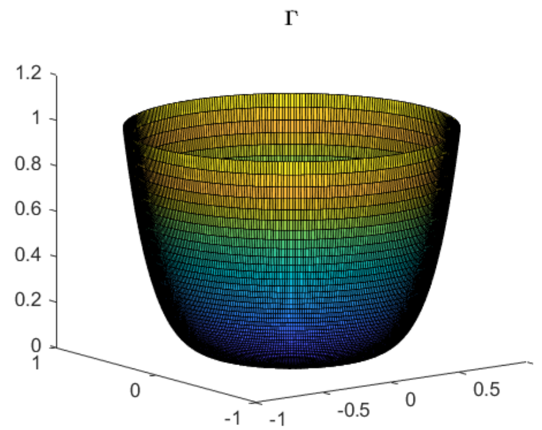
where $T_{\mathrm{qgoal}}\,(q)$=q is just the identity map

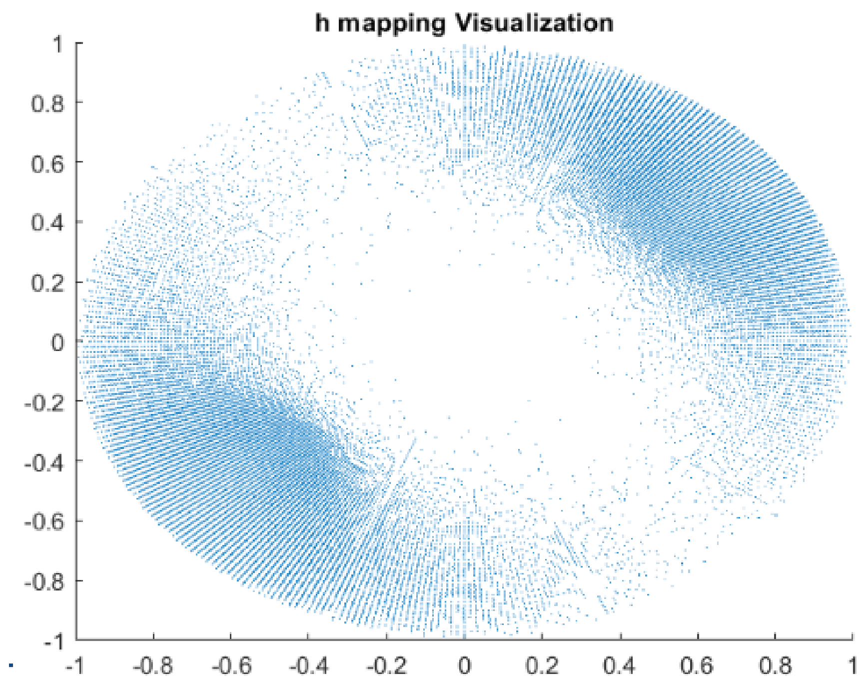
Parameter Description:

- Kappa: Associated with Attractive part of navigation function, **set as 3**.
- Lambda: Associated with analytical switch function ,**set as 1**.

Testing:







Inferences:

- We can observe here that there are two humps in beta visualization, signifying two obstacles.
- From the 2nd plot, we can observe the attractive part of navigation function, having a smooth minimum at the goal location.
- From the 3rd plot, we observed the h mapping between the star- and sphere-spaces and distribution of potential in the space and guaranteed to have a single minimum at q_goal for a sufficiently large κ.

To get the code:

Contact @ adityashah2310@gmail.com