ECE 209AS.1 Computational Robotics

Fall Quarter–2024

Path Planning Algorithms

Documented by:

RENISH ISRAEL (UID: 606530590)
First Year M.S. Student
Samueli Electrical and Computer Engineering
University of California Los Angeles

November 18, 2024

1 Artificial Force Field

Key Points

1. Also called "Artificial Potential Field" Algorithm (APF)
2. Robot navigates using "attractive" forces towards the goal or destination, and "repulsive" forces away from the obstacles
3. Very common and useful approach to avoid obstacles
4. The destination has the lowest potential, and the obstacles have the highest. Intuitively, the robot moves from higher to lower potentials (in the direction of decreasing gradient of the potential field)
5. A potential field is a physical field that obeys Laplace's equation. In other words, the force field associated with such a potential field ϕ can be evaluated by taking the gradient of the potential field.
$$\vec{F} = -\nabla\phi$$

Attractive Force

The potential field corresponding to the goal can be given as:

$$\phi_{\text{goal}}(x, y) = c\sqrt{(x - x_{\text{goal}})^2 + (y - y_{\text{goal}})^2}$$

, where (x, y) is the coordinate of the robot (node) and $(x_{\text{goal}}, y_{\text{goal}})$ is the coordinate of the 'goal'. c is a positive constant. Also, the force field can then be obtained as:

$$\vec{F}_{\text{goal}} = \left(\frac{-c}{\sqrt{(x - x_{\text{goal}})^2 + (y - y_{\text{goal}})^2}}\right)\left((x - x_{\text{goal}})\,\hat{\imath} + (y - y_{\text{goal}})\,\hat{\jmath}\right)$$

The formulation is such that the potential field ϕ_{goal} has a minima at $(x_{\text{goal}}, y_{\text{goal}})$

Repulsive Forces

Two kinds of repulsive forces are considered here, produced by the following sources:

1. The boundaries of the considered environment
2. All the obstacles in the environment

Boundary Potential:
The repulsive potential field ϕ_b caused by the boundaries can be formulated as follows:

$$\phi_b(x, y) = \frac{1}{\delta + \sum_{i=1}^{s}(g_i + |g_i|)}$$

, where δ is a small positive number to avoid ϕ_b going to ∞, s is the total number of boundary segments, and g_i is the linear equation that represents the boundary i mathematically.

Clearly, one can observe that there is a maxima for ϕ_b at each boundary.

Obstacle Potential:
Let the repulsive potential field generated by obstacle k is given by ϕ_k. If the obstacle k is a square of side l_k with center at (x_k, y_k), then ϕ_k is formulated as:

$$\phi_k(x, y) = \frac{\phi_{\max}}{1 + h(x, y)}$$

, where $h(x, y) = (x_k - l/2 - x) + |x_k - l/2 - x| + (y_k - l/2 - y) + |y_k - l/2 - y| + (x + 1 - x_k - l/2) + |x + 1 - x_k - l/2| + (y + 1 - y_k - l/2) + |y + 1 - y_k - l/2|$
Also, $\phi_{\max}$ is the maxima of the function ϕ_k for all values of k. The function h captures the side of the obstacle boundary the robot is in, and assigns potential accordingly. Finally, the obstacle potential ϕ_{obs} can be calculated as follows:

$$\phi_{\text{obs}} = \max\{\phi_k\}_{k=1}^{K}$$

, where K is the total number of obstacles in the environment.

Robot Trajectory

Now, the net force on the robot at time t can be calculated as follows:

$$\vec{F}_{\text{net}} = \vec{F}_{\text{goal}} - (\nabla\phi_b + \nabla\phi_{\text{obs}})$$

Hence, the robot's subsequent direction of motion should be given by the following direction vector: $\hat{d} = \cos\theta\,\hat{i} + \sin\theta\,\hat{j}$, where

$$\theta = \tan^{-1}\left(\frac{\vec{F}_{\text{net}} \cdot \hat{j}}{\vec{F}_{\text{net}} \cdot \hat{i}}\right)$$

References

1. https://medium.com/@rymshasiddiqui/path-planning-using-potential-field-algorithm-a30ad12bdb08
2. Robotic Planning: Chapter 4 by *Howie Choset*, CMU
3. Science Direct
4. https://youtu.be/Ls8EBoG_SEQ