Robotics Group Project - 5CCS2RGP

Lecture 9: Real-time Planning-Potential Field

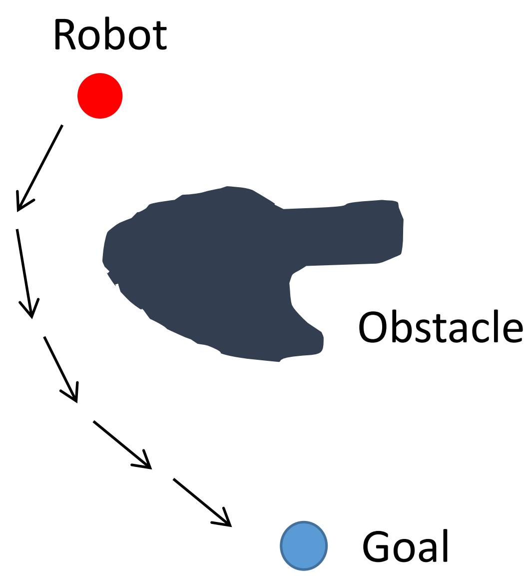
Potential Field Method

Artificial Potential Fields (APF)

A really simple idea:

– Suppose the goal is a point $g \in \mathfrak{R}^2$
– Suppose the robot is a point $r \in \mathfrak{R}^2$

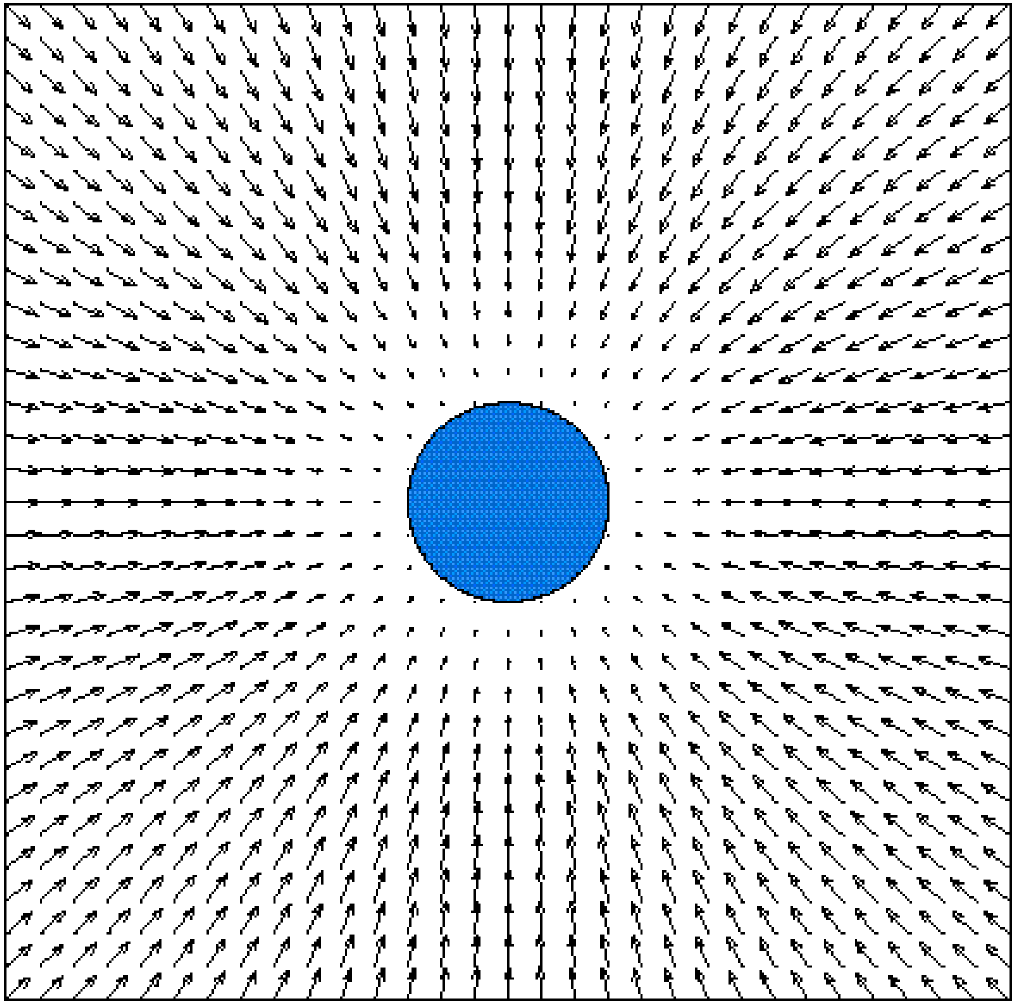
Think of a "spring" drawing the robot toward the goal and away from obstacles



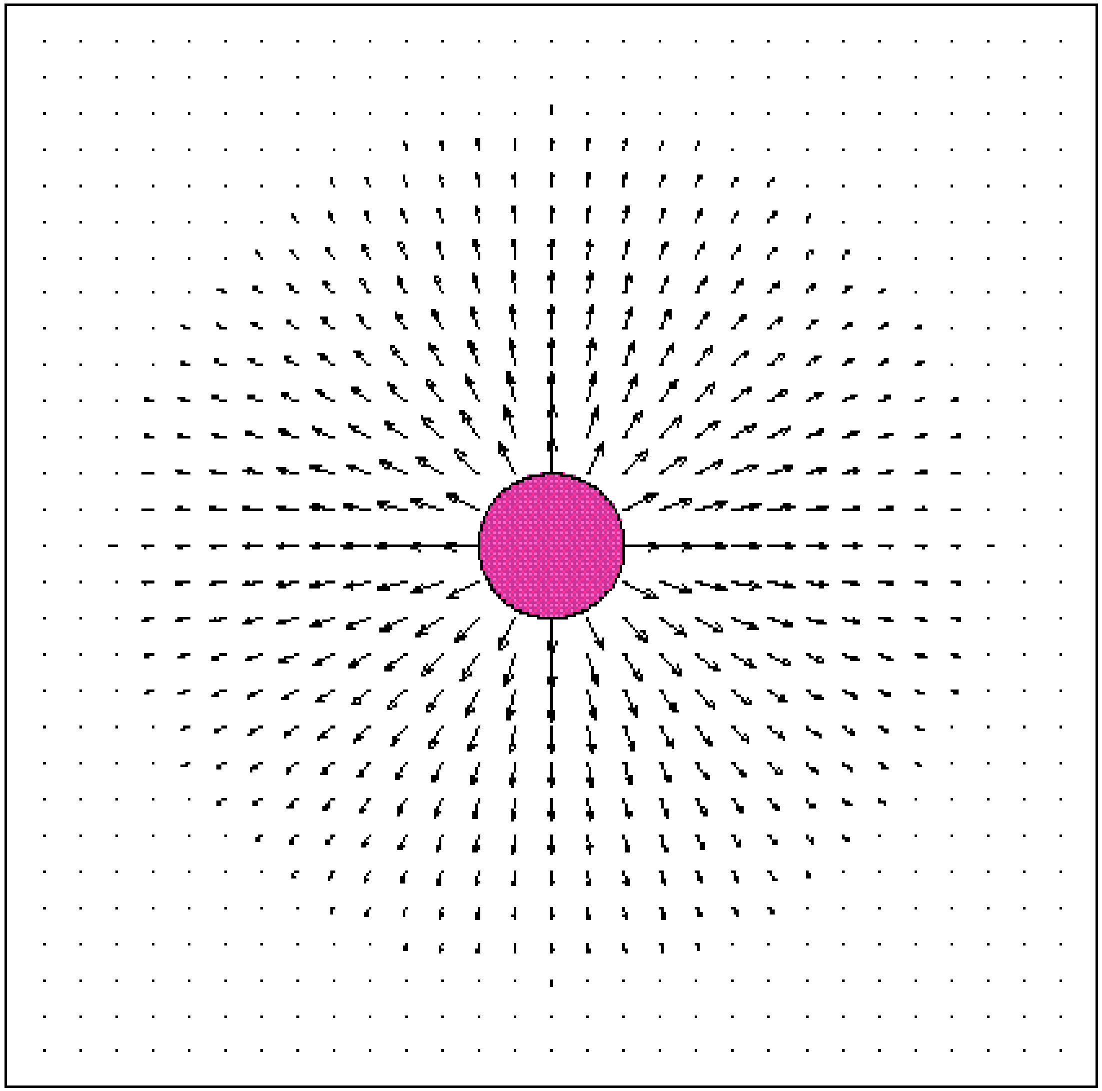
Potential Field Method

- Both the bowl and the spring analogies are ways of storing potential *energy*
- The robot moves to a lower energy configuration

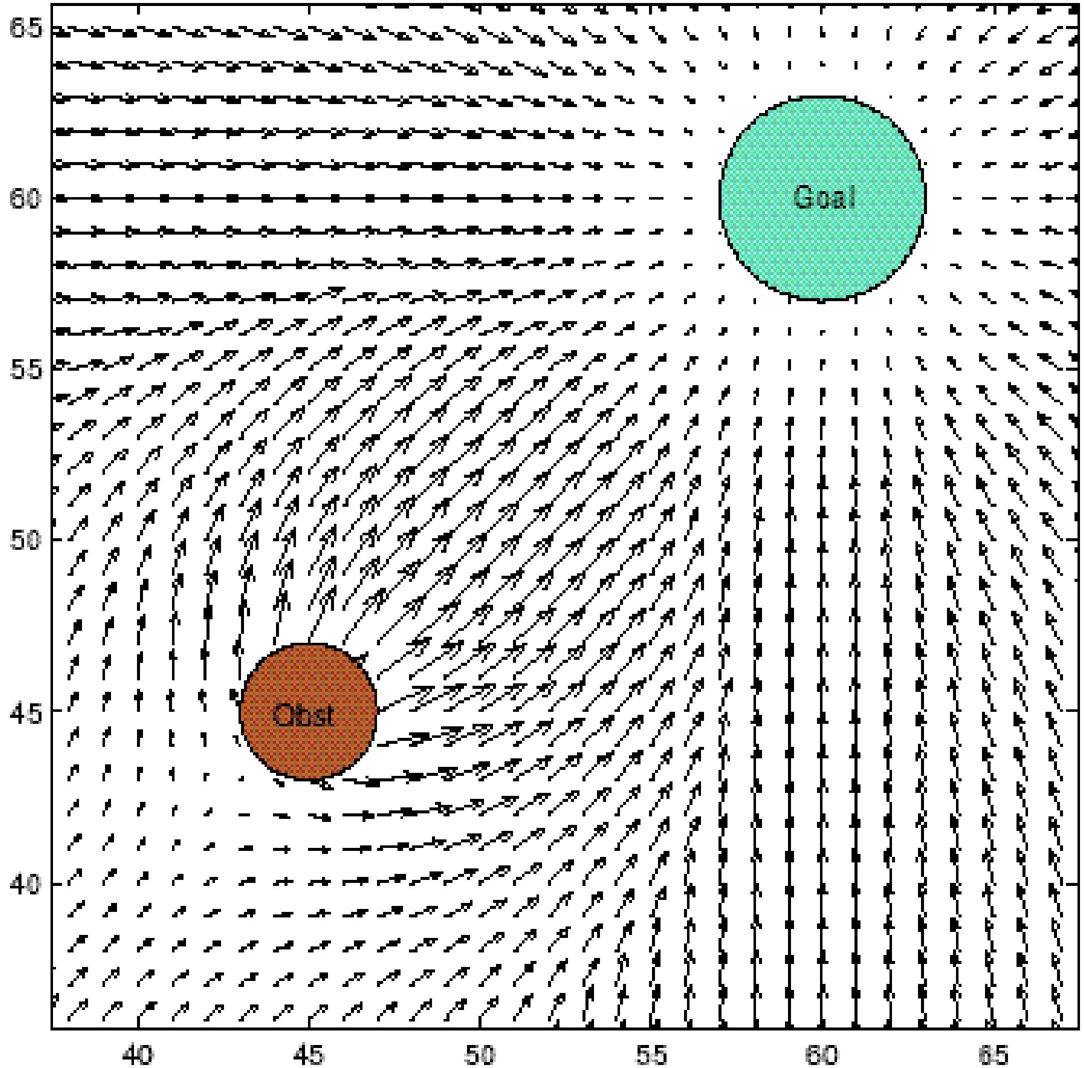
Attractive Potential Field



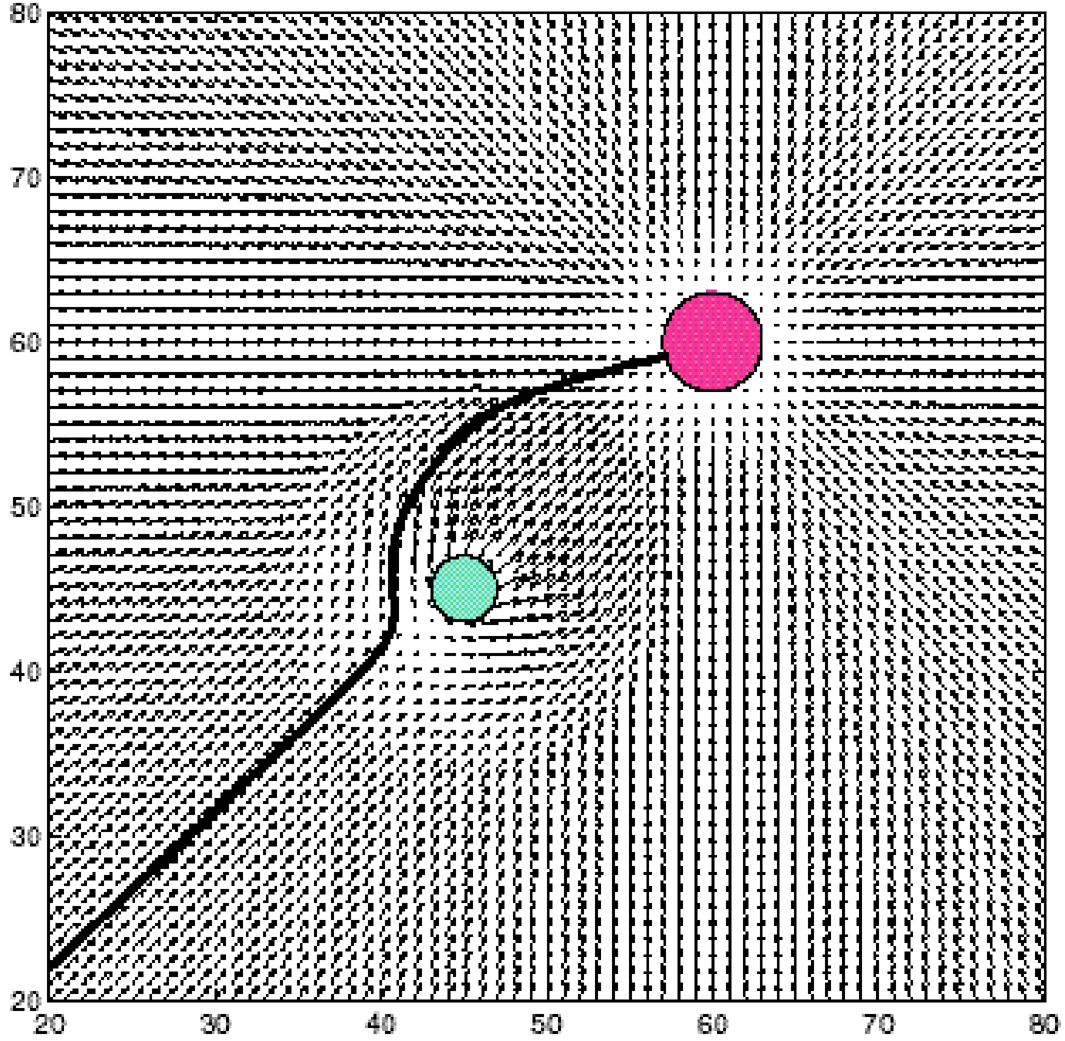
Repulsive Potential Field



Vector Sum of Two Fields



Resulting Robot Trajectory

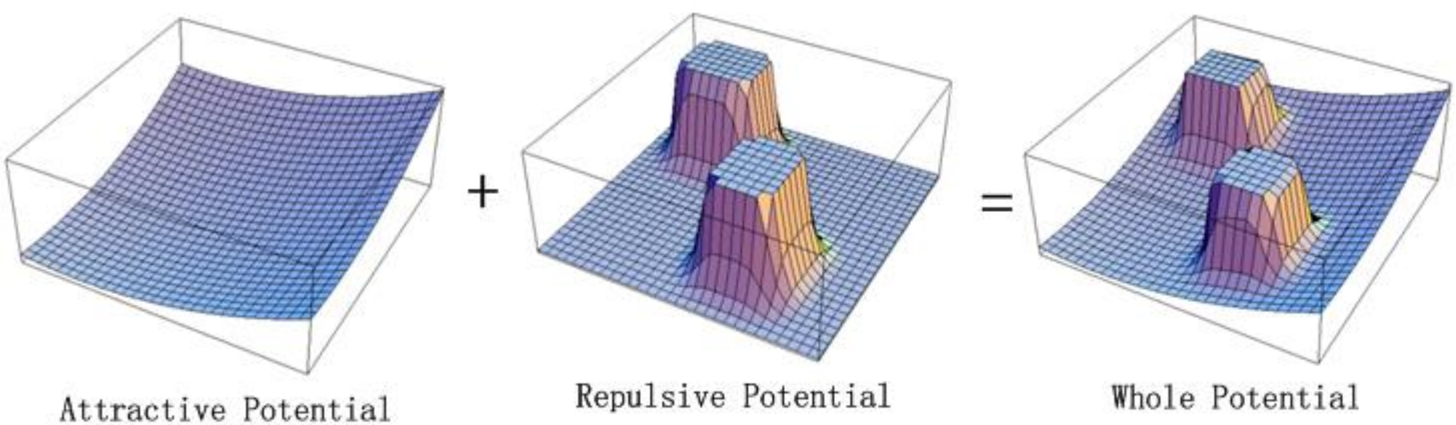


- A *potential function* is a function $U : \Re^{m} \rightarrow \Re$
- Bodies want to minimise their potential energy.
- Hence, they move **down** the potential energy gradient.
- Potential energy gradient: $\nabla U = \begin{bmatrix} \frac{\partial U}{\partial x} \\ \frac{\partial U}{\partial y} \end{bmatrix}$ (in 2D)
- This is equivalent to a **force** acting in the direction of the **negative** gradient (i.e. move down gradient)

$$\mathbf{F} = -\nabla U$$

Attractive/Repulsive Potential Field
– U_{att} is the "attractive" potential --- move to the goal
– U_{rep} is the "repulsive" potential --- avoid obstacles

$$U = U_{att} + U_{rep}$$



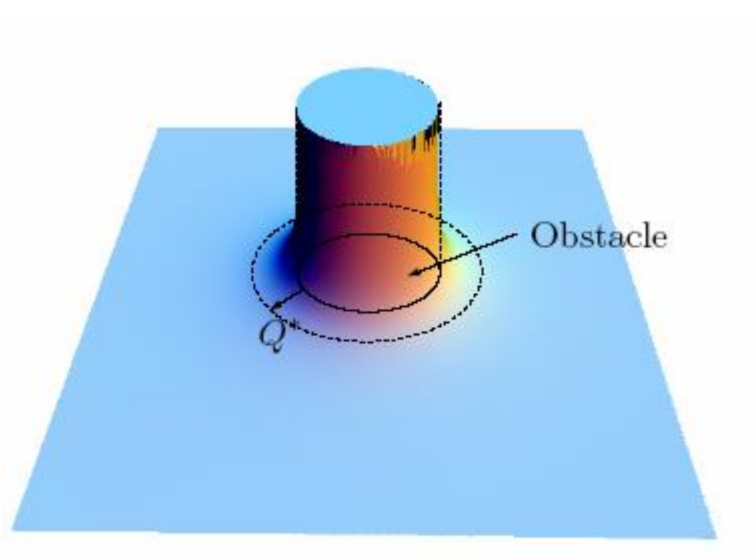
We can define the following potential functions (others are available!)

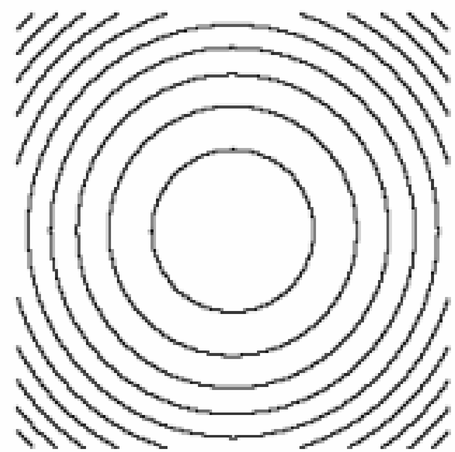
- Define: $D(\mathbf{p}_i) = \|\mathbf{p}_{robot} - \mathbf{p}_i\|$

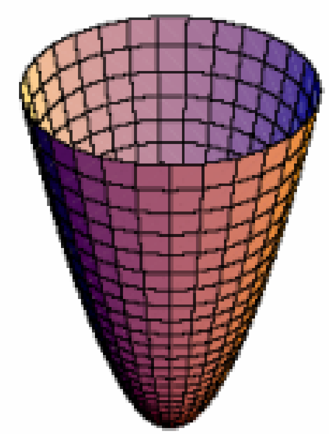
- $U_{att} = \frac{1}{2} K_{att} D^T\left(\mathbf{p}_{goal}\right) D$

- $U_{rep} = \begin{cases} \frac{1}{2} K_{rep} \left(\frac{1}{D(\mathbf{p}_{obs})} - \frac{1}{Q^*}\right)^2, D(\mathbf{p}_{obs}) < Q^* \\ 0, \qquad D(\mathbf{p}_{obs}) \geq Q^* \end{cases}$

- K_{att} and K_{rep} are positive constants chosen by user. They change the relative magnitude of the field.
- Q^* is the radius within which the repulsive potential is “activated” and can influence the robot. Value is chosen by user.







We calculate the following terms

- $D(\mathbf{p}_i) = \|\mathbf{p}_{robot} - \mathbf{p}_i\|$

- $\nabla U_{att} = K_{att} D\left(\mathbf{p}_{goal}\right) \nabla D\left(\mathbf{p}_{goal}\right)$

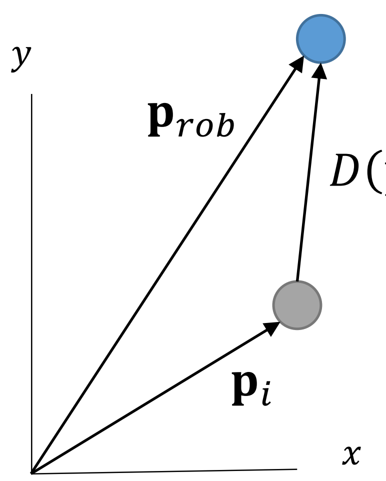
- $\nabla U_{rep} = \begin{cases} -K_{rep}\left(\frac{1}{D(\mathbf{p}_{obs})} - \frac{1}{Q^*}\right)\frac{1}{D^2(\mathbf{p}_{obs})}\nabla D(\mathbf{p}_{obs}), & D(\mathbf{p}_{obs}) < Q^* \\ 0, & D(\mathbf{p}_{obs}) \geq Q^* \end{cases}$

- $\nabla D(\mathbf{p}_i) = \begin{bmatrix} \frac{\partial D}{\partial x} \\ \frac{\partial D}{\partial y} \end{bmatrix}$

- $D(\mathbf{p}_i) = \|\mathbf{p}_{rob} - \mathbf{p}_i\|$
- $\|\mathbf{p}_{rob} - \mathbf{p}_i\| = \sqrt{(x_{rob} - x_i)^2 + (y_{rob} - y_i)^2}$

- $\frac{\partial D}{\partial x} = \frac{x_{rob}-x_i}{\sqrt{(x_{rob}-x_i)^2+(y_{rob}-y_i)^2}} = \frac{x_{rob}-x_i}{\|\mathbf{p}_{rob}-\mathbf{p}_i\|}$

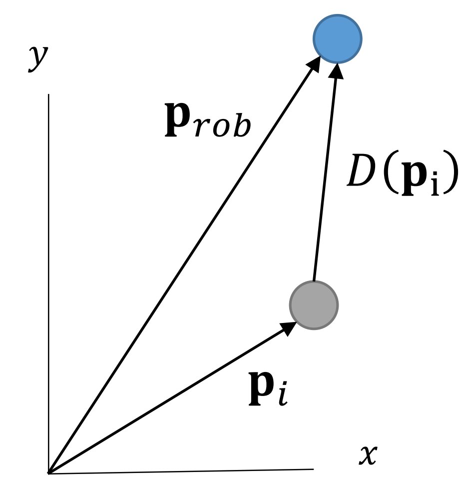
- $\frac{\partial D}{\partial y} = \frac{y_{rob}-y_i}{\sqrt{(x_{rob}-x_i)^2+(y_{rob}-y_i)^2}} = \frac{y_{rob}-y_i}{\|\mathbf{p}_{rob}-\mathbf{p}_i\|}$



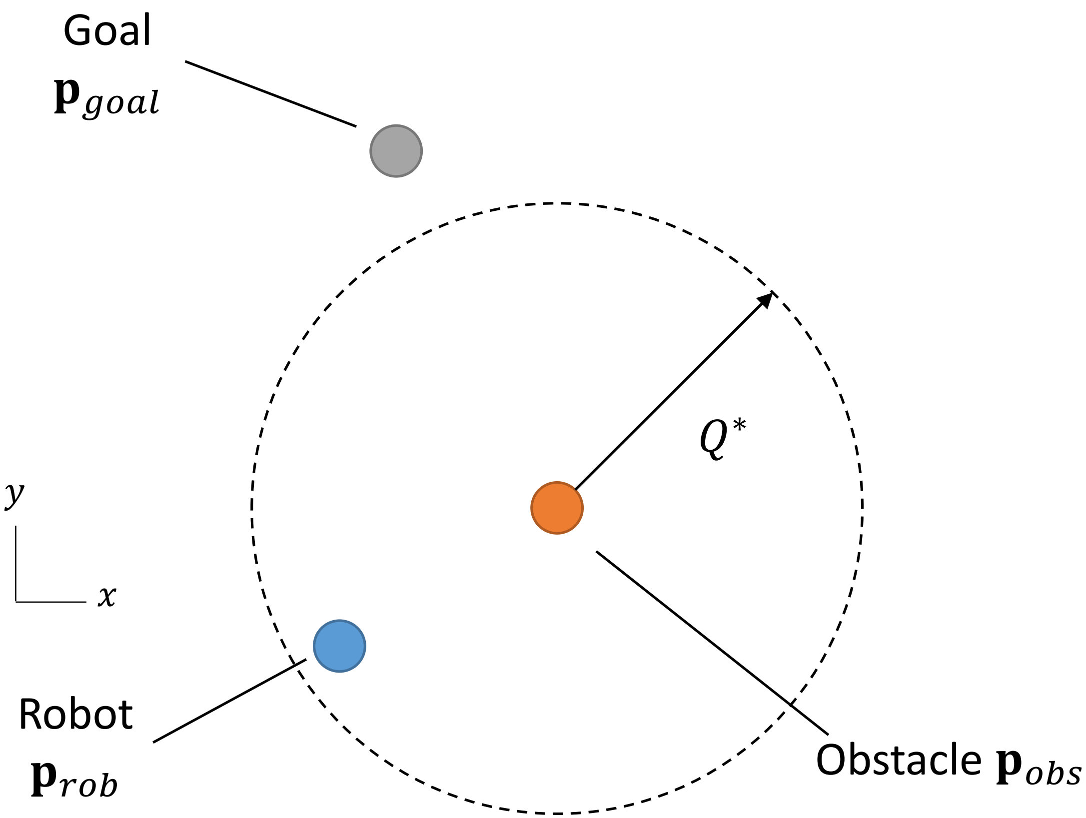
We can see that $\nabla D(\mathbf{p}_i)$ is just unit vector along line from $\mathbf{p}_i$ to $\mathbf{p}_{rob}$

$$\nabla D(\mathbf{p}_i) = \frac{1}{\|\mathbf{p}_{rob} - \mathbf{p}_i\|}\begin{bmatrix} x_{rob} - x_i \\ y_{rob} - y_i \end{bmatrix} = \frac{(\mathbf{p}_{rob} - \mathbf{p}_i)}{\|\mathbf{p}_{rob} - \mathbf{p}_i\|}$$

$$\nabla D(\mathbf{p}_i) = \frac{1}{D(\mathbf{p}_i)}(\mathbf{p}_{rob} - \mathbf{p}_i)$$



Example



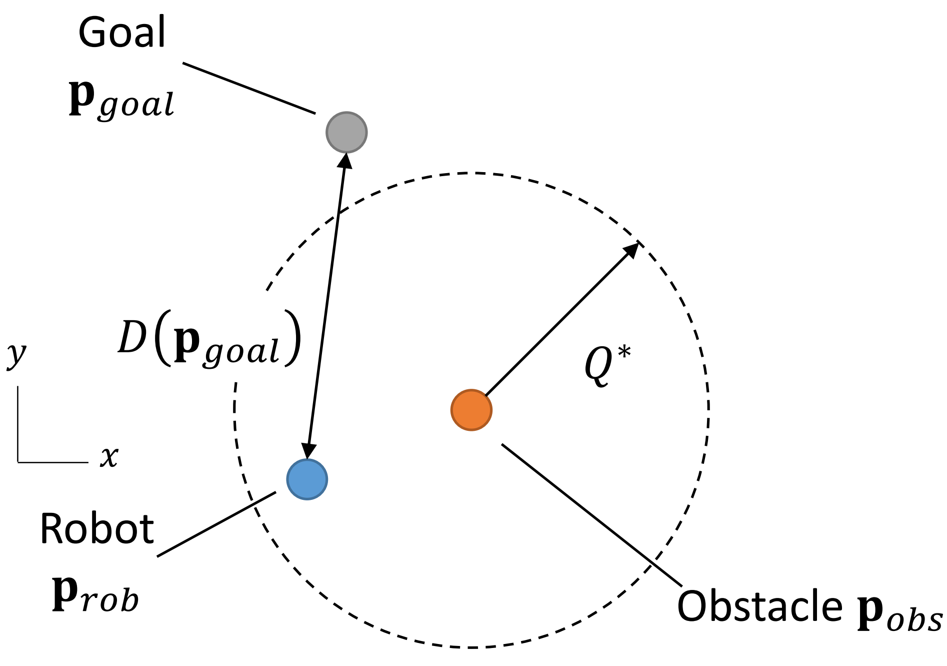
Let

$$\mathbf{p}_{goal} = [2 \quad 6]^{\mathrm{T}}$$
$$\mathbf{p}_{rob} = [1 \quad 1]^{\mathrm{T}}$$
$$\mathbf{p}_{obs} = [3 \quad 2]^{\mathrm{T}}$$
$$Q^* = 2.5$$
$$K_{att} = 1$$
$$K_{rep} = 100$$

Example

$$U_{att} = \frac{1}{2} K_{att} D^2(\mathbf{p}_{goal})$$

$$\nabla U_{att} = K_{att} D(\mathbf{p}_{goal}) \nabla D(\mathbf{p}_{goal})$$

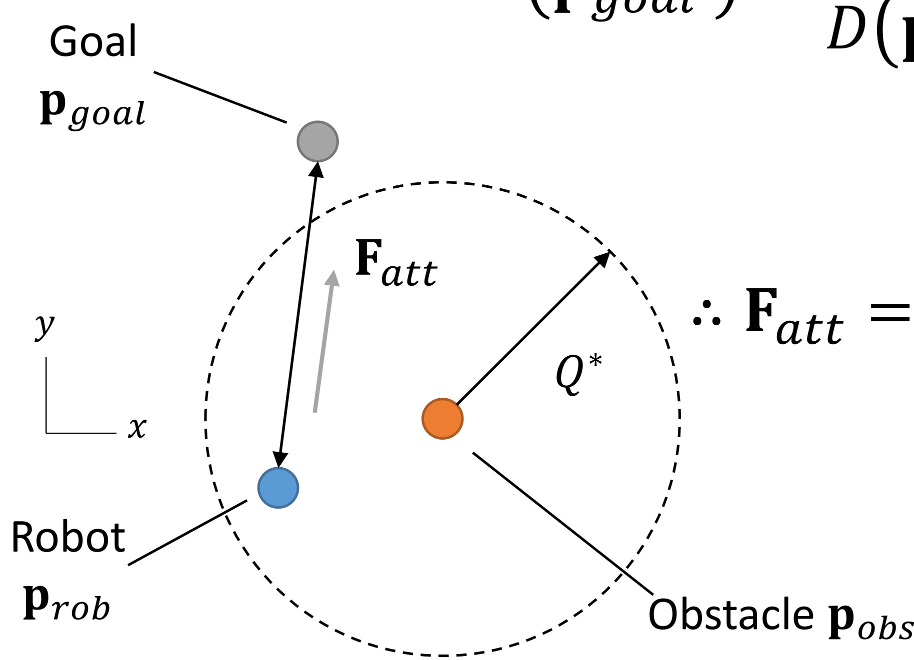


Calculating Attraction

- Hence,

$$\nabla U_{att} = K_{att} D(\mathbf{p}_{goal}) \nabla D(\mathbf{p}_{goal})$$

$$\nabla D(\mathbf{p}_{goal}) = \frac{1}{D(\mathbf{p}_{goal})}(\mathbf{p}_{rob} - \mathbf{p}_{goal})$$

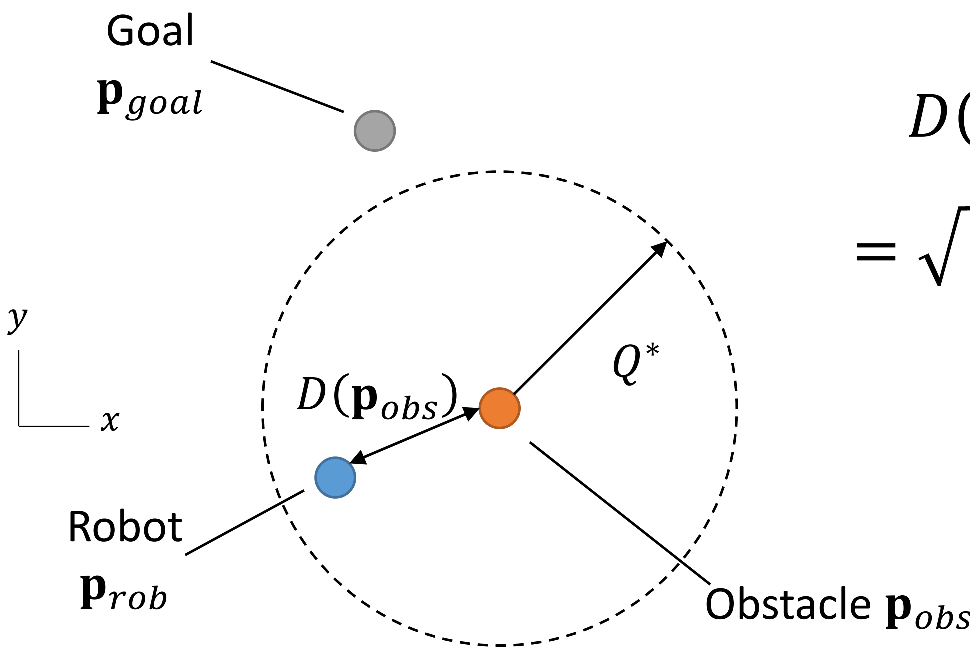


(Recall $F = -\nabla U$)

$$\therefore \mathbf{F}_{att} = -\nabla U_{att} = K_{att}(\mathbf{p}_{goal} - \mathbf{p}_{rob})$$

$$= (1)\left(\begin{bmatrix}2\\6\end{bmatrix} - \begin{bmatrix}1\\1\end{bmatrix}\right) = \begin{bmatrix}1\\5\end{bmatrix}$$

Calculating Repulsion



Is robot close enough to obstacle?

$$D(\mathbf{p}_{obs}) < Q^*?$$

$$D(\mathbf{p}_{obs}) = \|\mathbf{p}_{rob} - \mathbf{p}_{obs}\|$$

$$= \sqrt{(3-1)^2 + (2-1)^2} = 2.2$$

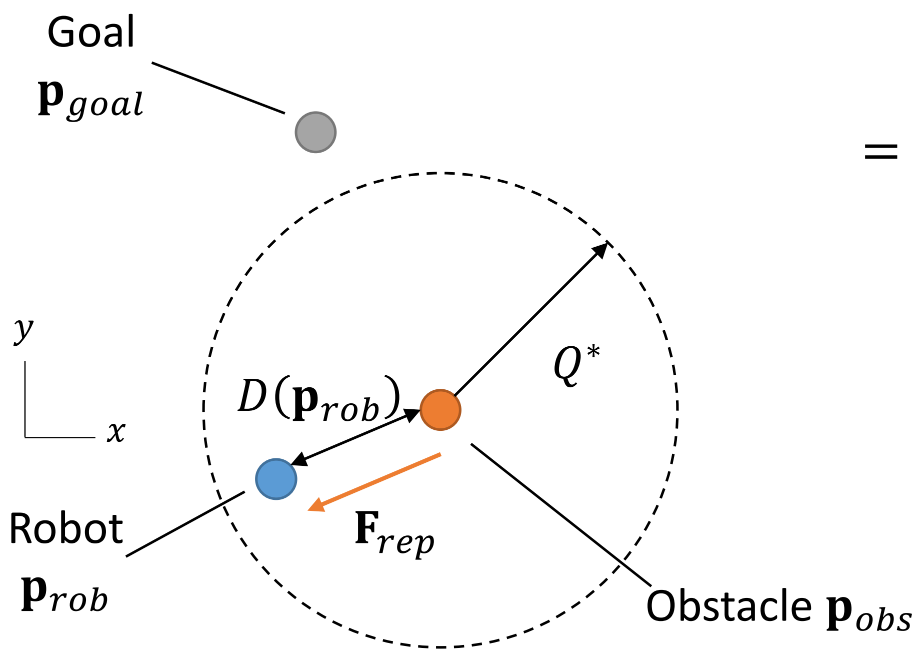
$$D(\mathbf{p}_{obs}) = 2.2 < 2.5$$

Yes

Calculating Repulsion

$$\nabla U_{rep} = -K_{rep}\left(\frac{1}{D(\mathbf{p}_{obs})} - \frac{1}{Q^*}\right)\frac{1}{D^2(\mathbf{p}_{obs})}\nabla D(\mathbf{p}_{obs})$$

$$\mathbf{F}_{rep} = -\nabla U_{rep} = K_{rep}\left(\frac{1}{D(\mathbf{p}_{obs})} - \frac{1}{Q^*}\right)\frac{1}{D^3(\mathbf{p}_{obs})}(\mathbf{p}_{rob} - \mathbf{p}_{obs})$$



$$= (100)\left(\frac{1}{2.2} - \frac{1}{2.5}\right)\frac{1}{2.2^3}\left(\begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}3\\2\end{bmatrix}\right)$$

$$\mathbf{F}_{rep} = \begin{bmatrix}-1.02\\-0.53\end{bmatrix}$$

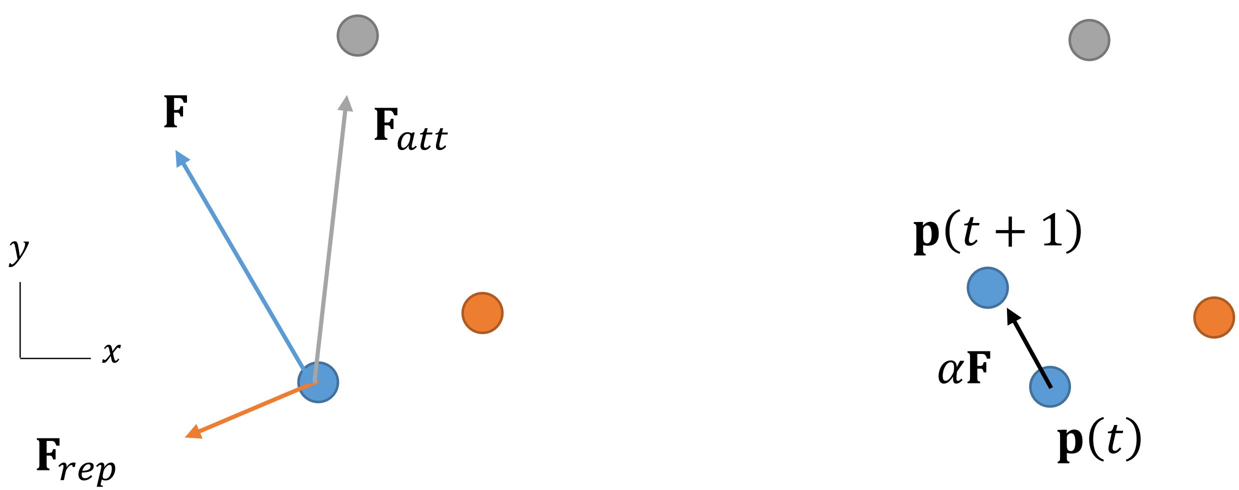
Calculating Total Force

$$\mathbf{F} = -\nabla U = -(\nabla U_{att} + \nabla U_{rep}) = \mathbf{F}_{att} + \mathbf{F}_{rep}$$

Can use this to find next position when performing a discrete simulation:

$$\mathbf{p}_{rob}(t+1) = \mathbf{p}_{rob}(t) + \alpha\mathbf{F},$$

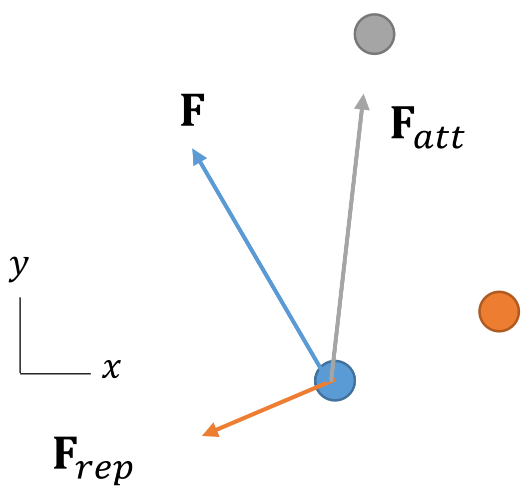
α: scalar constant ("step size"). This is a tuning parameter and can be any value chosen by user to fit situation

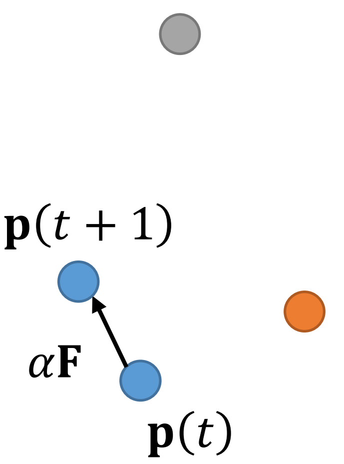


Calculating Movement (for simulation)

$$\alpha = 0.1$$

$$\mathbf{p}(t+1) = \mathbf{p}(t) + \alpha\mathbf{F} = \begin{bmatrix}1\\1\end{bmatrix} + 0.1\left(\begin{bmatrix}-1.02\\-0.53\end{bmatrix} + \begin{bmatrix}1\\5\end{bmatrix}\right) = \begin{bmatrix}1.006\\2.341\end{bmatrix}$$

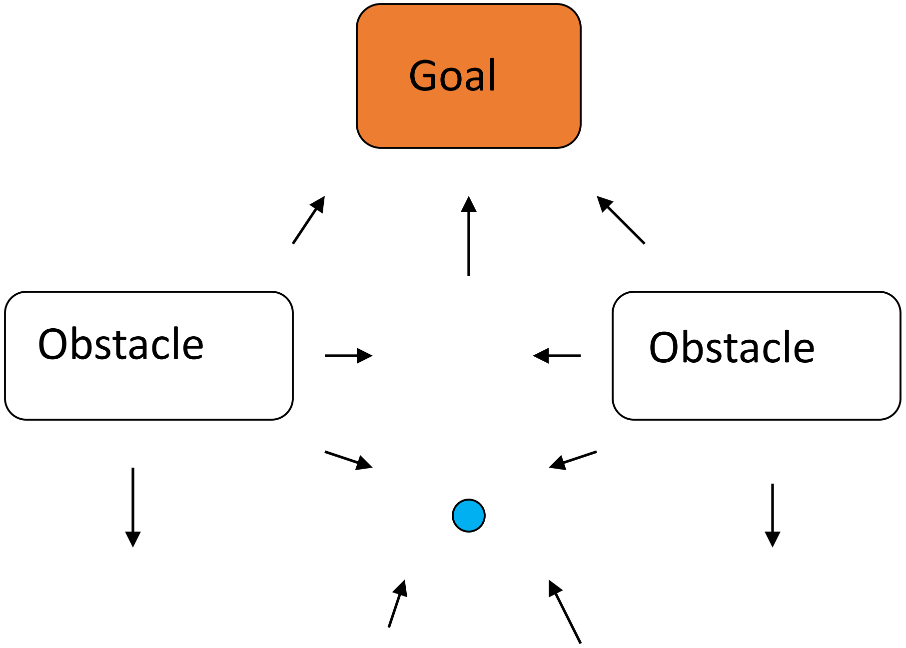




Potential Problems with Potential Fields

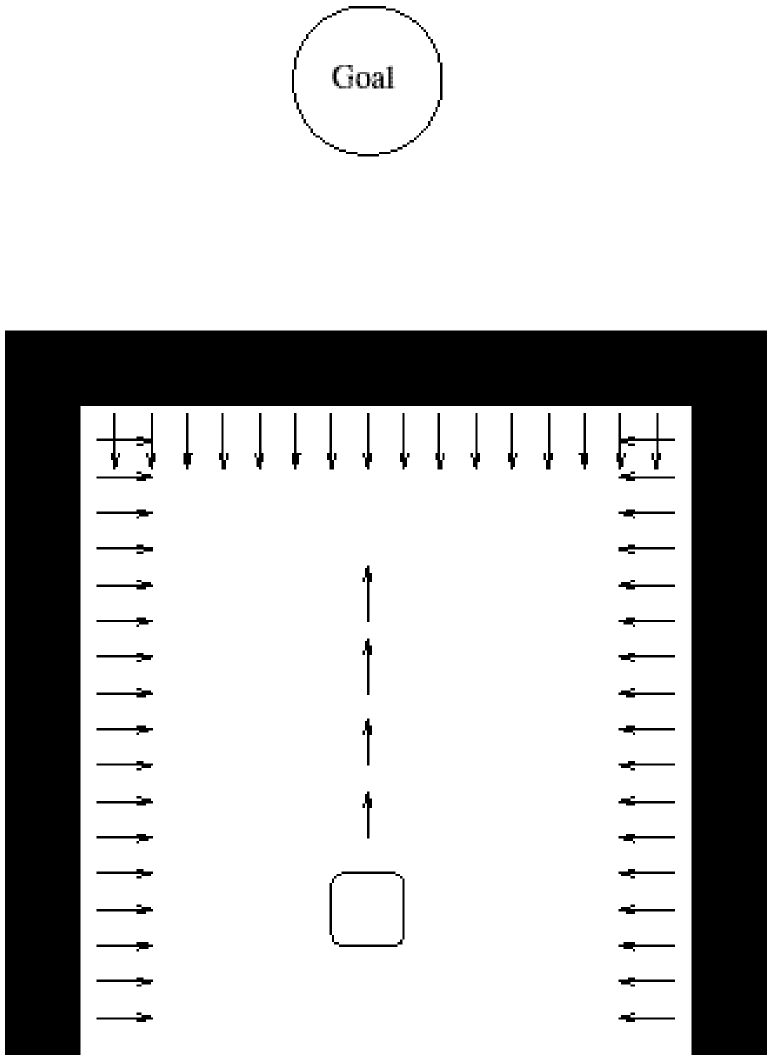
- *Local minima*
  - Attractive and repulsive forces can balance, so robot makes no progress.
  - Closely spaced obstacles, or dead end.
- *Unstable oscillation*
  - The dynamics of the robot/environment system can become unstable.
  - High speeds, narrow corridors, sudden changes.

Local Minimum Problem



Box Canyon Problem

- Local minimum problem, or
- *AvoidPast* potential field.



Rotational and Random Fields

- Not gradients of potential functions
- Adding a *rotational field* around obstacles
  - Breaks symmetry
  - Avoids some local minima
  - Guides robot around groups of obstacles
- A *random field* gets the robot unstuck.
  - Avoids some local minima.