Previously Feedback control

- Kinematics of the differential drive robot in polar coordinates
- Designed state feedback controller (error feedback)
- At each instance of time compute a control law
- Given the current error between current and desired position
Kinematic Position Control: Coordinates Transformation

Coordinates transformation into polar coordinates with its origin at goal position:

\[
\rho = \sqrt{\Delta x^2 + \Delta y^2}
\]

\[
\alpha = -\theta + \arctan2(\Delta y, \Delta x)
\]

\[
\beta = -\theta - \alpha
\]

System description, in the new polar coordinates

\[
\begin{bmatrix}
\dot{\rho} \\
\dot{\alpha} \\
\dot{\beta}
\end{bmatrix} =
\begin{bmatrix}
-cos(\alpha) & 0 \\
sin(\alpha) & -1 \\
-sin(\alpha) & 0
\end{bmatrix}
\begin{bmatrix}
\rho \\
\alpha \\
\beta
\end{bmatrix} +
\begin{bmatrix}
0 \\
1 \\
0
\end{bmatrix} v
\]

For \(\alpha\) from \(I_1 = \left(-\frac{\pi}{2}, \frac{\pi}{2}\right)\) for \(I_2 = (-\pi, -\pi/2) \cup (\pi/2, \pi]\)

---

Kinematic Position Control: The Control Law

- It can be shown, that with
  \[v = k_\rho \rho \quad \omega = k_\alpha \alpha + k_\beta \beta\]

the feedback controlled system

\[
\begin{bmatrix}
\dot{\rho} \\
\dot{\alpha} \\
\dot{\beta}
\end{bmatrix} =
\begin{bmatrix}
-k_\rho \rho \cos(\alpha) \\
-k_\rho \rho \sin(\alpha) - k_\alpha \alpha - k_\beta \beta \\
-k_\rho \rho \sin(\alpha)
\end{bmatrix}
\]

- will drive the robot to \((\rho, \alpha, \beta) = (0,0,0)\)
- The control signal \(v\) has always constant sign,
  - the direction of movement is kept positive or negative during movement
  - parking maneuver is performed always in the most natural way and without ever inverting its motion.
Potential Field Methods

- Idea robot is a particle, point with unit mass
- Environment is represented as a potential field (locally)
- Advantage – capability to generate on-line collision avoidance

Compute force acting on a robot – incremental path planning
q – the pose of the robot, here only position
\[ F(q) = -\nabla U(q) \quad q = [x, y]^T \]

Example: Robot can translate freely, we can control independently
Environment represented by a potential function
\[ U(x, y) \]

Force is proportional to the gradient of the potential function
\[ \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = -\nabla U(x, y) \]

Jana Kosecka, GMU
Attractive Potential Field

- Linear function of distance
\[ U_a(q) = \xi \| q - q_{goal} \| \quad F_a(q) = -\nabla U_a(q) = -\xi \frac{(q - q_{goal})}{\| q - q_{goal} \|} \]
\[ q = [x, y]^T \quad q_{goal} = [x_{goal}, y_{goal}]^T \]
- Quadratic function of distance
\[ U_a(q) = \xi \frac{1}{2} \| q - q_{goal} \|^2 \quad F_a(q) = -\nabla U_a(q) = -\xi (q - q_{goal}) \]

Combination of two – far away use linear, closer by use parabolic well

Repulsive Potential Field

\[ U_r(q) = \frac{1}{2} \nu \left( \frac{1}{\rho(q, q_{obst})} - \frac{1}{\rho_c} \right) \quad \text{if} \quad \rho(q, q_{obst}) \leq \rho_c \]
\[ \text{else} \quad U_r(q) = 0 \]

Minimal distance between the robot and the obstacle
\[ \rho(q, q_{obst}) = \| q - q_{obst} \| \]
Repulsive Potential Field

\[ U_{rep} = \frac{1}{2} \nu \left( \frac{1}{\rho(q, q_{obst})} - \frac{1}{\rho_0} \right)^2 \]

if \( \rho(q, q_{obst}) \leq \rho_C \)

else \( U_r(q) = 0 \)

Minimal distance between the robot and the obstacle

\[ F_{rep} = -\nabla U_{rep} = \nu \left( \frac{1}{\rho(q)} - \frac{1}{\rho_0} \right) \frac{1}{\rho(q)^2} \frac{q - q_{obs}}{\rho(q)} \]

Outside of sensitivity zone repulsive force is 0

Previously – repulsive potential related to the square of the Inverse distance – here just proportional to inverse distance

Note: need to compute gradient to get the force
Potential Fields

Resulting force  $F(q) = -\nabla (U_o(q) + U_r(q))$

Iterative gradient descent planning

$q_{i+1} = q_i + \delta_i \frac{F(q)}{\|F(q)\|}$

$q = [x, y]^T$

Potential Fields

- Simple way to get to the bottom, follow the gradient

$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = -\nabla U(x, y) \quad \dot{q} = -\nabla U(q)$

$q_{i+1} = q_i + \delta_i \frac{F(q)}{\|F(q)\|}$

- i.e. Gradient descent strategy  $\nabla U(q) = 0$

- A critical, stationary point is such that

- Equation is stationary at the critical point
**Computing Distances**

- In practice the distances are computed using sensors
- Obstacles are not circular
- Consider a range sensor which returns the closest distance to the obstacle

**Potential Functions**

- How do we know we have single global minimum?
- If global minimum is not guaranteed, need to do something else then gradient descent
- Design functions in such a way that global minimum can be guaranteed
Potential function

- Heuristics for escaping the local minima
- Can be used in local and global context
- Numerical techniques, Random walk methods

- Navigation functions (Rimon & Kodistchek, 92)
- Navigations in sphere worlds and worlds diffeomorphic to them


\[ \phi(q) = -\frac{d^2(q, q_{goal})}{\left[d(q, q_{goal})^{2k} + \beta(q)\right]^{1/k}} \beta(q) \]  

Obstacle term

- For sufficiently large k – this is a navigation function [Rimon-Koditschek, 92]
Potential Field Path Planning: Using Harmonic Potentials

• Hydrodynamics analogy
  – robot is moving similar to a fluid particle following

• Note:
  – Complicated, only simulation shown

Potential fields for Rigid Bodies

• So far robot was considered a point - gradient of the potential function – force acting on a point
• How to generalize to manipulators of objects?
• Idea – forces acting on objects – forces acting on multiple points of the object (blackboard)
• [http://www.cs.cmu.edu/~motionplanning/](http://www.cs.cmu.edu/~motionplanning/)
• For robots, pick enough control points to pin down the robot – define forces in workspace – map them to configuration space
Potential Fields Issues

- Local minima
- Heuristics for escaping the local minima
- Can be used in local and global context
- Numerical techniques
- Random walk methods
- To be continued