Potential field and controller

CS 470 Introduction To Artificial Intelligence

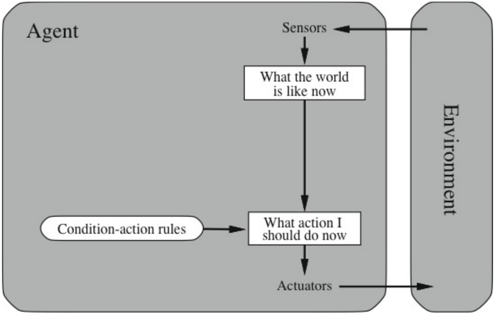
Daqing Yi

Department of Computer Science
Brigham Young University

Outline

Reactive robotics

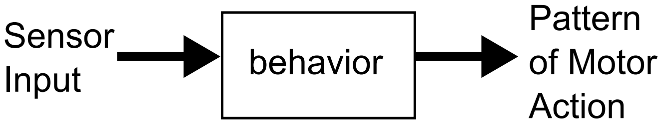
- "Intelligence without representation"
- Reflex agent



Reactive robotics

Behavior-based robotics

- activity-generating building block
- sensor input o
- action a
- $a = f(o)$



Reactive robotics

When we need a reflex agent

- Rule-based requirement
- Simple
- Fast response

Example: pick up something

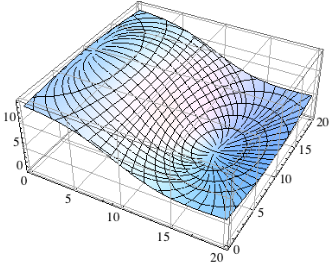
Potential field

potential

- energy distribution on the map

potential field

- gradient of the potential
- array (or field) of vector
- vector represents a force (magnitude and direction)

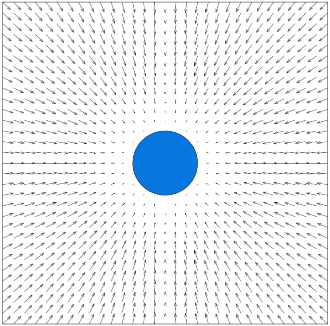


Attractive potential field

$$d = \sqrt{(x_G - x)^2 + (y_G - y)^2}$$

$$\theta = \arctan\left(\frac{y_G - y}{x_G - x}\right)$$

$$\begin{aligned}
\text{if } d < f \quad & \Delta x = \Delta y = 0 \\
\text{if } r \le d \le s + r \quad & \Delta x = \alpha(d - r)\cos(\theta) \\
& \Delta y = \alpha(d - r)\sin(\theta) \\
\text{if } d > s + r \quad & \Delta x = \alpha s \cos(\theta) \\
& \Delta y = \alpha s \sin(\theta)
\end{aligned}$$

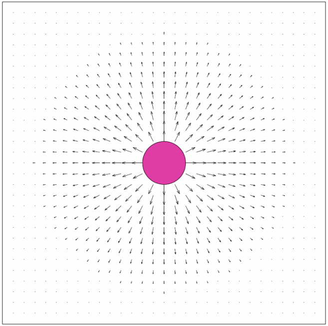


Repulsive potential field

$$d = \sqrt{(x_G - x)^2 + (y_G - y)^2}$$

$$\theta = \arctan\left(\frac{y_G - y}{x_G - x}\right)$$

$$\begin{array}{rl}
\text{if } d < f & \Delta x = -sign(\cos(\theta))\infty \\
& \Delta y = -sign(\sin(\theta))\infty \\
\text{if } r \leq d \leq s + r & \Delta x = -\beta(s + r - d)\cos(\theta) \\
& \Delta y = -\beta(s + r - d)\sin(\theta) \\
\text{if } d > s + r & \Delta x = \Delta y = 0
\end{array}$$

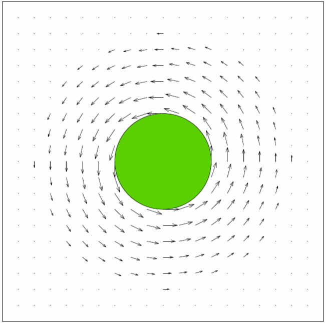


Tangential potential field

$$d = \sqrt{(x_G - x)^2 + (y_G - y)^2}$$

$$\theta = \arctan(\frac{y_G - y}{x_G - x})$$

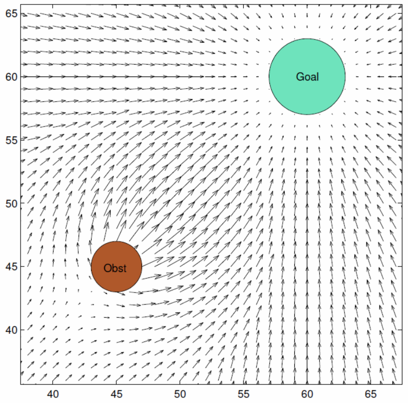
$$\begin{array}{rr}
\text{if } d < f & \Delta x = -sign(\cos(\theta \pm \frac{\pi}{2}))\infty \\
& \Delta y = -sign(\sin(\theta \pm \frac{\pi}{2}))\infty \\
\text{if } r \leq d \leq s + r & \Delta x = -\beta(s + r - d)\cos(\theta \pm \frac{\pi}{2}) \\
& \Delta y = -\beta(s + r - d)\sin(\theta \pm \frac{\pi}{2}) \\
\text{if } d > s + r & \Delta x = \Delta y = 0
\end{array}$$



Application

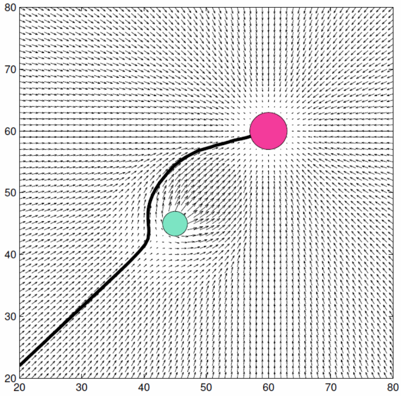
Merge potential field

Weighing the potential fields



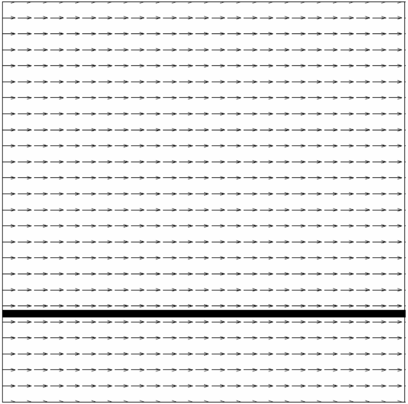
Application

path planning + obstacle avoidance



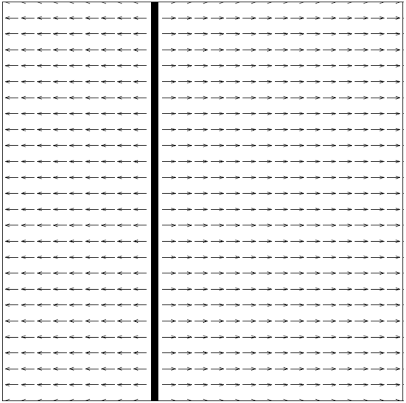
More types of potential fields

Uniform potential field



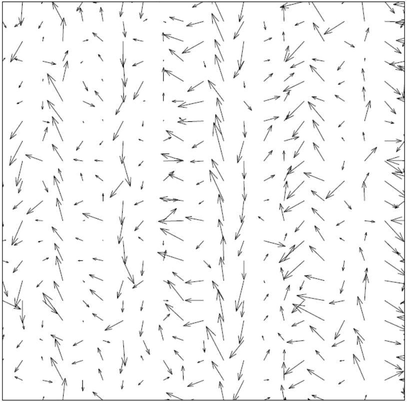
More types of potential fields

Perpendicular potential field

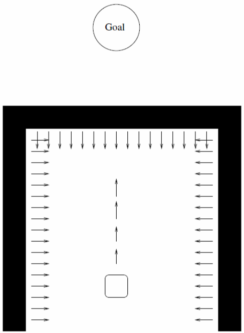


More types of potential fields

Random potential field

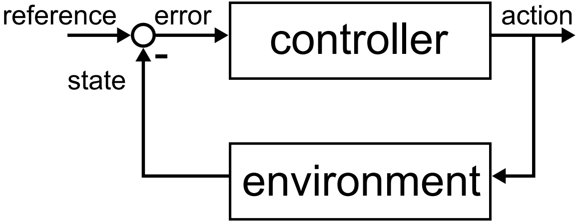


More types of potential fields



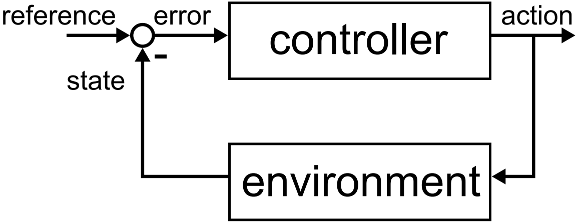
Controller

- error = reference - state
- action = controller (error)
- state = environment (action)



Controller

- Position reaching
- Trajectory tracking

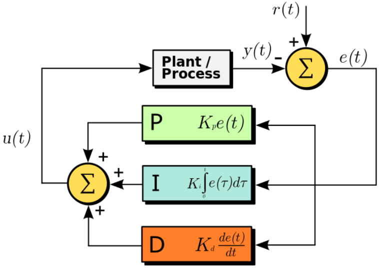


PID controller

PID controller

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t)$$

- **P**ropotional: $K_p e(t)$
- **I**ntegral: $K_i \int_0^t e(\tau) d\tau$
- **D**erivative: $K_d \frac{d}{dt} e(t)$



PID controller

Parameter tuning

- Propotional K_p
  - current error
  - too small $\longrightarrow$ long rise time
  - too big $\longrightarrow$ big overshoot
- Integral K_i
  - accumulated error
  - increase the precision
  - too small $\longrightarrow$ big overshoot
  - too big $\longrightarrow$ long setting time
- Derivative K_d
  - error variation
  - sensitive to noise
  - dependent on sampling interval

PID controller

- P controller
- PI controller
- PID controller