

# Práctica 7

## Evasión de obstáculos mediante campos potenciales artificiales

Laboratorio de Bio-Robótica  
Robots Móviles y Agentes Inteligentes

### Objetivos

- Implementar un comportamiento reactivo para evasión de obstáculos mediante campos potenciales artificiales.
- Determinar los parámetros de diseño para obtener un comportamiento satisfactorio.
- Probar la evasión de obstáculos tanto en simulación como experimentalmente.

## 1. Marco Teórico

### 1.1. Campos potenciales artificiales

Los campos potenciales artificiales son un método para evadir obstáculos de manera reactiva, es decir, no es necesario que dichos obstáculos estén en algún tipo de representación del espacio. En el caso de A\*, por ejemplo, se pueden trazar rutas que evadan obstáculos pero estos deben estar definidos en el mapa de celdas de ocupación a partir del cual se calculan dichas rutas.

Los campos potenciales artificiales, como su nombre lo indica, son un método en el que se diseña una función potencial tal que, si el robot “desciende” por dicha función, seguirá una ruta que lo llevará al punto meta al mismo tiempo que evade los obstáculos a su paso.

Una función potencial es una función real diferenciable  $U : R^n \rightarrow R$  que puede verse como una función de energía y que, por lo tanto, su gradiente  $\nabla U(q)$ , donde  $q$  es la posición del robot, representa fuerza.

El gradiente de la función, dado por

$$\nabla U(q) = \left[ \frac{\partial U}{\partial q_1}, \dots, \frac{\partial U}{\partial q_n} \right] \quad (1)$$

es un vector que apunta en la dirección de máximo cambio de  $U$ . Si esta función potencial se diseña de modo que tenga un mínimo en la posición meta y máximos locales en la posición de cada obstáculo que se desee evadir, entonces se puede mover al robot mediante el algoritmo de descenso del gradiente y esto hará que el robot continúe moviéndose hasta encontrar un punto  $q^*$  en el cual  $\nabla U(q^*) = 0$ .

## 1.2. Descenso del gradiente

## 1.3. El sensor láser Hokuyo-URG

# 2. Tareas

## 2.1. Prerrequisitos

Antes de continuar, actualice el repositorio y recompile:

```
cd ~/RoboticsCourses
git pull origin master
cd catkin_ws
catkin_make
```

## 2.2. Nodo que implementa los campos potenciales

# 3. Evaluación

- El código debe estar ordenado.
- **Importante:** Si el alumno no conoce su código, NO se contará la práctica.