

Exploración con mapeo dinámico

Fernando Carmona, Damián De Biase, Ramiro Verrastro, Rodolfo Cignoli
Grupo de Inteligencia Artificial y Robótica (GIAR) UTN-FRBA, Buenos Aires, Argentina
cfcarmona22@yahoo.com.ar, damiandebiase@yahoo.com.ar

Resumen-En los robots móviles, la necesidad de detectar obstáculos y evadirlos es una tarea fundamental. En este trabajo se presenta un sistema que, además de ser capaz de detectar y evadir obstáculos, genera un mapa del entorno, el cual le permite recalcular la trayectoria obteniendo el camino mas corto hasta el objetivo.

I. INTRODUCCIÓN

El objetivo de este trabajo es lograr que un robot se desplace de un punto a otro en un entorno desconocido, siguiendo una trayectoria calculada y evadiendo los obstáculos que vaya descubriendo en la misma, generando además, un mapa de dicho entorno. Para lograr este objetivo, las acciones que realiza el sistema son:

- Detección de obstáculos
- Cálculo de la distancia al obstáculo
- Ubicación del obstáculo en el mapa
- Generación de trayectoria

El sistema esta compuesto físicamente por una cámara web y un dispositivo láser montado sobre un motor (Fig. 1), que a su vez se encuentra sobre un robot móvil ER1 [1] que cuenta con tracción de tipo diferencial (Fig. 2). El robot lleva una notebook, en la cual se ejecuta un programa servidor que realiza todo el procesamiento y control del mismo. El usuario se comunica con este programa desde otra maquina, mediante una red wi-fi y le envía las instrucciones a ejecutar desde un programa cliente diseñado para tal efecto. Ambos programas están realizados en Visual Studio.NET [2].



Figura 1. Conjunto cámara-láser-motor.



Figura 2. Robot ER1

II. ESTADO DEL ARTE

Para la generación de mapas existen diferentes modelos. Los mapas topológicos se basan en relacionar geoméricamente las características del entorno, Buschka [3] presenta un sistema que detecta habitaciones y realiza la topología del entorno. Por otro lado, dentro de los mapas métricos están los que trabajan con elementos de modelos geoméricos [4], y los de rejillas de ocupación [5], en el cual existen distintos paradigmas para la creencia de ocupación, como son el modelo probabilístico bayesiano [6] y el enfoque histográfico [7].

Dentro de las técnicas para la evasión de obstáculos, se pueden citar: los campos potenciales artificiales [8], la cual se basa en que el robot se mueve inmerso en campos potenciales generados por la meta (potenciales de atracción) y por los obstáculos (potenciales de repulsión) y otros como el histograma de campos vectoriales, y el de las bandas elásticas descriptos en el trabajo de Maria Isabel Ribeiro [9].

En este trabajo se utiliza un mapa métrico de rejillas de ocupación, ya que es adecuado para la fusión de datos de diferentes sensores y para la evasión de obstáculos se utiliza una implementación del algoritmo A* (léase A estrella) [10] [11] [12], el cual es muy utilizado en video juegos (Pathfinding) [13].

III. DESARROLLO

Para que un robot autónomo pueda desempeñarse en un ambiente desconocido, es necesario que obtenga datos del entorno con relativa precisión, los interprete y utilice para la toma de decisiones. Estos datos pueden tener distinto origen y ser obtenidos mediante diferentes medios.

A continuación se explica de forma simplificada como funciona el mapeo durante el desplazamiento del robot. Una vez definido el punto de origen y el de destino, el robot comienza a moverse siguiendo una trayectoria calculada, la cual es una línea recta en el caso de que no haya datos previos. A medida que se desplaza, el sistema "observa" en la dirección del movimiento, mediante la rotación o "paneo" del conjunto cámara y láser. Los objetos detectados se dibujan en el mapa; cuando uno de estos objetos aparece en el camino y se

encuentra a una distancia definida como mínima, el robot se detiene y completa el paneo.

Una vez completado el paneo, el sistema recalcula el camino (esta vez desde la posición en la que se encuentra), considerando todos los obstáculos detectados durante el desplazamiento anterior y obtiene nuevamente el trayecto mas corto, dentro del mapa conocido hasta el momento.

Este proceso es repetido la cantidad de veces que sea necesario hasta alcanzar el punto de destino.

A. *Detección de obstáculo y cálculo de la distancia*

Mediante la utilización de una cámara no es posible obtener información de profundidad en la imagen, razón por la cual se utiliza un punto láser como referencia en la imagen (*Laser Range Finder*) [14]. En la Fig. 3 se puede observar un esquema del sistema.

Figura 3. Esquema del sistema

Figura 4. Modelo del sistema

El eje que pasa por el centro de la cámara es paralelo al haz del láser, y en la imagen capturada por la cámara aparece el punto láser junto con el resto de la escena (Fig. 4). Cada vez que se

Los valores obtenidos de rpp y off son los siguientes:

rpp = 0,00249 rad/pixeles
off = 0,03999 rad

Finalmente, para cada nueva imagen adquirida es posible calcular la distancia a un objeto mediante la formula (4),

$$D = h / \text{tg}(\text{pci} \cdot \text{rpp} + \text{off}) \quad (4)$$

- Obtener el camino mas corto
- Obtener el camino en el menor tiempo posible

Lo primero a definir en la implementación del algoritmo es el área de búsqueda, en este caso es una matriz bidimensional, en la que cada elemento representa uno de los cuadrados de la grilla, y se almacena su estado con un valor de probabilidad de ocupación.

Figura 10. El robot confirma la existencia del obstáculo y recalcula su trayectoria.

Figura 11. El robot detecta un nuevo obstáculo y realiza el mismo procedimiento.

V. CONCLUSIONES

El cálculo de la distancia al obstáculo con el sistema propuesto dio resultados satisfactorios para la generación de un mapa de ocupación si m y

ara

ancia mayores 635(a)-2()636(e)-2(s)3(t)1(a)142(s)3()6-3(e)-2(l)1()634(s)3(i)1(s)3(t)1(e)-14(m)6(a)-2()634(s)3ea ueld

-6(s)-2, su 1362(a)-2ddcud

ome 1123(e)-14(n)6-2(1123(e)-14(n)6(c)-2(o))18(n)6* ((r)-5aa)-2(r)-5-2(1123(e)-(l)1()1123(c)-2(a)-14(m-6(i)1n)-6(o)-6-2(124(hu)6ae)-2(s)3(t)1(a)manca para la 1003()TJ 11.211 0 Td [(e)-14(n)6(s)3(i)1(d)-6a0

-9-v)63ea)-2(l)1(o)-6(c)-2(i)1(d)-6a0 d pano dl siste