

# Maapeo y navegaci3n por nodos de potencial para rob3tica m3vil

<sup>1,1</sup> A. J. Carimatto, <sup>1,2</sup> C. Verrastro, <sup>1,3</sup> J.C. G3mez

<sup>1</sup>Grupo de Inteligencia Artificial y Rob3tica, UTN-FRBA, Buenos Aires, Argentina,

<sup>2</sup>CNEA, Centro At3mico Ezeiza

<sup>3</sup>Instituto Nacional de Tecnologa Industrial (INTI).

[carimatto@cae.cnea.gov.ar](mailto:carimatto@cae.cnea.gov.ar), [cverra@cae.cnea.gov.ar](mailto:cverra@cae.cnea.gov.ar), [juanca@inti.gov.ar](mailto:juanca@inti.gov.ar)

**Resumen:** Se presenta un m3todo de mapeo y navegaci3n para robot m3vil que por medio de sensores pueda desenvolverse tanto en ambientes conocidos como desconocidos. Se realiza una representaci3n del mapa f3sico como un 3rbol de nodos que caracterizan los recintos, que ser3n los hitos para encontrar el camino al objetivo.

**Palabras Clave:** navegaci3n, mapeo, localizaci3n, robot m3vil, campos el3ctricos.

## 1. INTRODUCCION

El presente trabajo es parte de un proyecto del Grupo de Inteligencia Artificial y Rob3tica de UTN-FRBA, orientado a dotar a robots m3viles de capacidad de navegar y operar en ambientes semiestructurados.

Existen numerosos m3todos para resolver el problema de encontrar un camino al objetivo en un mapa dado, entre estos se destacan:

- Algoritmo A\* (A- Estrella) aplicado al mapa: su ventaja principal es la obtenci3n del camino m3nimo, pero como desventaja tiene un elevado costo de c3lculo y se obtiene un camino poligonal.

- M3todos aplicando funci3n potencial (atractivo y repulsivo): tienen relativa facilidad de c3lculo y logran un camino suave; en contraparte, existen situaciones en las que no encuentran soluci3n por quedar atrapado el robot en un m3nimo de potencial. [6]

- Algoritmos gen3ticos [7]

- Algoritmos Bug: El algoritmo es simple de aplicar, sin embargo, el m3todo es muy rudimentario, consiguiendo caminos ineficientes[6].

En todos los casos, el mapa carece de interpretaci3n, es s3lo una grilla de puntos.

Tomando como referencia los trabajos mencionados. El trabajo pretende resolver:

- B3squeda de camino en un laberinto dinámico.

- Suavizado del camino asegurando que el robot no haga movimientos bruscos.

- Estructuraci3n del mapa en sectores de inter3s en un grafo o mapa abstracto.

A trav3s de la experiencia con los distintos m3todos, se observ3 que aqu3llos que utilizan funci3n potencial obtenían trayectorias m3s suaves y naturales, a pesar de presentar como desventaja problemas de explosi3n combinatoria y de m3nimos locales en mapas complejos. Sin embargo los m3nimos aparecen agrupados en lugares estrat3gicos del mapa. Se propone hacer uso de esta característica agrupando los puntos en zonas, asignándole a cada

zona un punto característico y luego utilizarlos como hitos para encontrar entre ellos la mejor combinaci3n para llegar al objetivo.

## 2. M3TODOS

Como primer paso, se hace una recolecci3n de informaci3n y obtenci3n de un mapa simplificado (mapeo), esto se realiz3 con un sensor ultras3nico, colocado sobre una base m3vil que gira a velocidad constante.

Con la grilla en bruto se realiza una serie de operaciones para caracterizar el mapa y para encontrar recintos y puntos de inter3s; con 3stos se podr3 conformar un 3rbol de nodos o hitos que servir3 para encontrar recorridos posibles a un objetivo. El m3todo consiste en convertir un mapa f3sico, en un mapa de cargas, calcular el potencial el3ctrico en todo el mapa, hallar puntos singulares, agruparlos en nodos o hitos, armar el 3rbol de caminos y aplicar el algoritmo A\* para conseguir la mejor combinaci3n de hitos entre el origen y el destino que minimice el camino recorrido, en las secciones siguientes se describen los pasos en detalle.

### 2.1. B3squeda del camino en un laberinto dinámico: M3todo Potencial\*

#### 2.1.1 Conversi3n matriz de puntos a cargas

Se dispone de una grilla cargada con el laberinto a estudiar (completo o no) como en la figura 1, cada porci3n de pared est3 representada por una carga positiva, al igual que el punto de partida del robot (c3rculo), y una carga negativa para la meta (cruz).

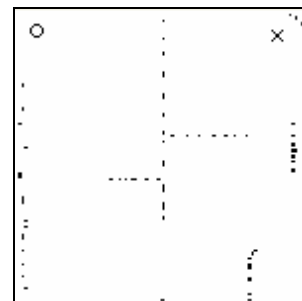


Figura 1: Distribuci3n de cargas

#### 2.1.2 Puntos singulares

Con la matriz de cargas, se calcula el potencial el3ctrico en todos los puntos (x;y) de la grilla como:

$$V_{(x,y)} = \sum_{i=0}^n \frac{K}{|(x,y) - (x_i, y_i)|^2} \quad (1)$$

Donde,  $(x_i, y_i)$  es la coordenada de cada carga y K es la constante de proporcionalidad

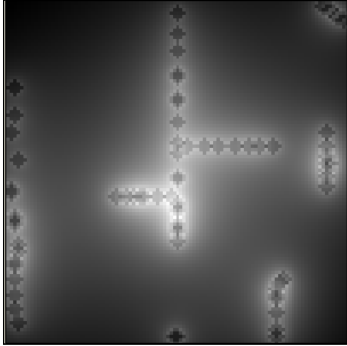


Figura 2: Mapa de potencial eléctrico dónde el nivel de gris representa la tensión  $V_{(x,y)}$

Se adopta como definición de punto singular en  $(x,y)$  a aquellos que en alguna dirección posee sus dos vecinos con mayor potencial. Estos puntos concuerdan con las aperturas de los recintos, y en general a cualquier punto intermedio entre dos paredes o cargas de la misma polaridad.



Figura 3: Puntos singulares

### 2.1.3. Segmentación

Una vez obtenidos los puntos de mínimo potencial, se segmentan por cercanía; a cada segmento se le calcula su centro de masa. De esta manera se obtiene un nuevo conjunto de puntos característicos o nodos, conformando un mapa abstracto o grafo del entorno. Se puede pensar que cada nodo del grafo resultante representa un recinto del mapa real. Para el caso de una habitación cerrada, el nodo estará en el centro de la habitación; si la habitación tiene una puerta abierta, el nodo se desplaza hacia la abertura. Asimismo, habrá mínimos que caractericen aberturas, puertas, pasillos; recordando que éstos existen siempre entre dos cargas enfrentadas de la misma polaridad. Entonces cada recinto o región de interés queda representado por un nodo.

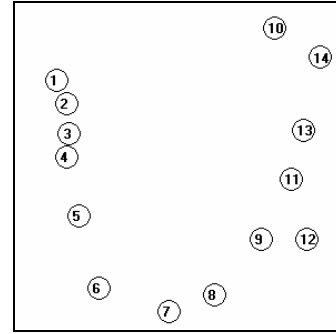


Figura 4: Nodos del árbol de caminos. La cantidad depende del sistema de segmentación

El proceso empleado en la segmentación es similar a la detección de Blobs, el algoritmo recorre la grilla; al encontrar un punto singular, se presentan dos alternativas:

- Si existe otro punto singular a una distancia menor a  $r$  (parámetro ajustable relacionado con el diámetro del robot) se asigna al mismo grupo.

- Si no existe otro punto singular aledaño, o existe pero no posee grupo asignado, se crea uno nuevo.

Ejemplo: dada la grilla de puntos la figura 15.

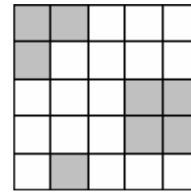


Figura 15: Grilla con puntos singulares

Se comienza a analizar la coordenada  $(1,1)$ , existe un punto singular aledaño, pero sin grupo asignado, se creará uno nuevo.

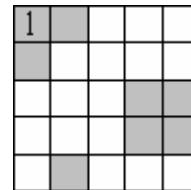


Figura 16: grilla de segmentación después de analizar el punto  $(1,1)$

El siguiente punto a procesar es el  $(2,1)$ , éste punto tiene puntos aledaños con grupo asignado, se agruparán con él.

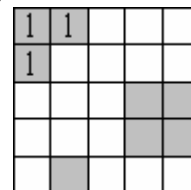


Figura 17: grilla de segmentación después de analizar las primeras dos filas

El algoritmo sigue recorriendo la grilla hasta obtener finalmente:

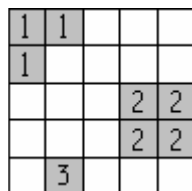


Figura 18: Todos los puntos tienen asignados un grupo.

Luego, se calcula el centro de masa de cada grupo y se obtiene la coordenada del nodo característico, como se ve en la figura 19

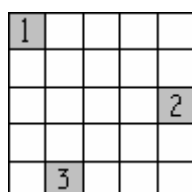


Figura 19: Grilla de nodos después de la segmentación

#### 2.1.4. Armado de la red o caminos

Este conjunto de puntos, en el que están incluidos los puntos representativos del mapa se utiliza para encontrar el camino mínimo. Aplicando el algoritmo A\*, se encontrará la mejor combinación de ellos para lograr el objetivo, previo armado de la matriz de costos (distancias) entre todos los puntos. Este algoritmo queda supeditado a la interconexión de los nodos. Entre dos nodos cualesquiera puede comprobarse la “visibilidad” entre ambos teniendo en cuenta la matriz original de cargas.

Una posibilidad de resolución de problema de la visibilidad es imaginar al par de puntos como dos cargas opuestas y trazar el campo eléctrico, además se debe tener en cuenta la distancia entre ruedas para verificar el paso. Independientemente de la función de visibilidad que se adopte, utilizamos esta información para truncar los caminos sin paso libre del algoritmo A\*. Con estos datos se busca la mejor combinación de puntos que hace que el robot llegue a destino Figura 5a.

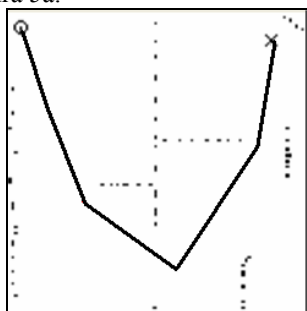


Figura 5a: Recorrido obtenido por el robot

#### 2.1.5. Suavizado del camino

Una vez encontrado el camino, se procede a su recálculo utilizando el método de las líneas de campos eléctricos (figura 5b). Se asigna a cada punto del recorrido y el siguiente (puntos A y B de la figura 6) cargas opuestas; aparecerán campos eléctricos que comenzarán en A y terminarán en B. De todo el conjunto de líneas de campo, se selecciona a aquella que parte desde A con un ángulo igual (o al menos similar) al ángulo con el que el robot llegó a dicho punto y verifica la condición de paso.

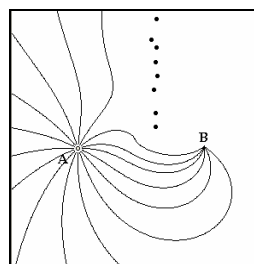


Figura 6: Caminos posibles entre dos nodos cualesquiera (A y B)

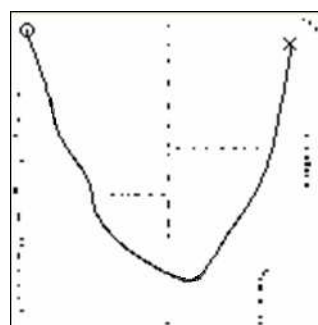


Figura 5b: Recorrido obtenido por el robot luego de aplicar el método de suavizado.

#### 2.2 Traslado al sistema físico

Teniendo la trayectoria, la última etapa es calcular el movimiento de las ruedas necesario para que el robot describa las curvas que lo llevarán a destino. Para tal fin, la trayectoria se visualiza como una sucesión de puntos; en cada punto se conoce la orientación y posición que deberá tener el robot, calculándose los arcos de trayectoria que son necesarios.

#### 2.3. Trazado del mapa utilizando sensor ultrasónico (mapeo dinámico)

Este apartado trata el mecanismo para utilizar este método en situaciones donde no se conoce el mapa con anterioridad, por el contrario, la tarea es recoger los datos. Teniendo como únicos datos la posición del robot y el destino o posición a la que se desea llegar.

La primer tarea a resolver es, desde la posición inicial, realizar un mapeo del entorno y guardar esa información tal como se describe en [5]. En la siguiente etapa se aplica el método Potencial\*, que

encuentra una ruta de acción teniendo en cuenta el mapa conocido hasta el momento. El robot inicia el desplazamiento sobre la ruta planificada. Durante el desplazamiento el sensor detecta nuevos obstáculos y los vuelca a la grilla que representa el mapa. El robot móvil seguirá su camino aún cuando aparezcan nuevos obstáculos en el mapa, siempre que no se encuentren en la ruta planificada; si este fuera el caso, el robot recalcula la ruta y el proceso se repite hasta llegar al objetivo.

El sensor toma mediciones en todos los ángulos, cada medición representada por su ángulo y módulo respecto del robot deberá transformarse a coordenadas respecto del mapa.

$$x_0 = x_r * m_m * \cos(\alpha_r + \alpha_m)$$

$$y_0 = y_r * m_m * \sin(\alpha_r + \alpha_m)$$

$(x_0; y_0)$  : coordenadas de la celda

$(m_m; \alpha_m)$  : módulo y ángulo de la medición

$(x_r; y_r; \alpha_r)$  ; posición y ángulo del robot

La celda  $(x_0; y_0)$  de la grilla se reforzará, aumentando su carga positiva, por el contrario, todos los puntos que estén en el segmento definido por  $(x_r; y_r)$  y  $(x_0; y_0)$  se le debilita la carga. De esta manera, el mapa se actualiza dinámicamente. La velocidad de cambio del entorno, no puede superar a la velocidad de recolección de datos del sensor. Este método fue implementado con un sonar, pero puede ser extendido a otros sensores de distancia como sensores Laser, radar o cámaras de video o fusión de ellos.

## 2.4. Un caso real

A continuación, se analizará un caso real, en él se verá el resultado de los mecanismos explicados. Para realizar esta prueba, se colocó una serie de planchas de madera en el recinto a modo de paredes y se ubicó al robot dotado de un sensor ultrasónico en el recinto interior. Se bordeó al mapa de cargas, definiendo un entorno de acción acotado. Luego de realizado el mapeo, se ve el entorno descrito mediante las cargas eléctricas (Figura 7). Luego aplicando el algoritmo Potencial\*, se encontró el camino a la meta. El algoritmo generó 14 nodos, 12 de ellos que están numerados de 1 a 12 y dos más que corresponden a la salida (marcado con O) y a la meta (marcado con S). Aplicando A\* sobre dichos nodos se obtuvo como mejor solución a la combinación O/4/7/S. Otras soluciones posibles arrojadas por el algoritmo fueron: O/4/5/7/S, O/2/3/4/7/11/S, etc...

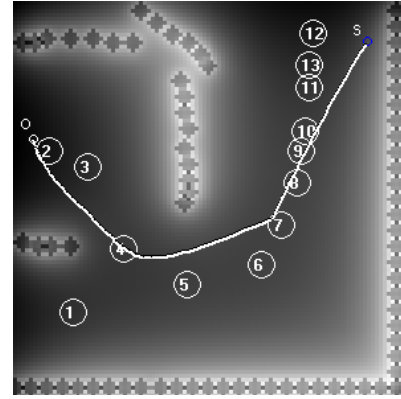


Figura 7: Grilla de distribución de cargas, árbol de nodos y el camino hallado.

## 2.5. Detalles del software de navegación

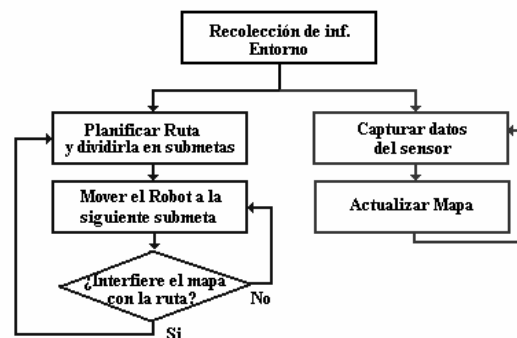


Figura 8: Esquema del software de navegación

El diagrama de la figura 8 representa dos hilos de ejecución concurrentes, el de la izquierda se encarga de la navegación, el de la derecha se encarga de la recolección de información y armado del mapa.

## 3. CONCLUSIONES

El método descrito disminuye el tiempo de procesamiento para hallar un camino al objetivo con respecto a la aplicación directa del algoritmo A\*, reduciendo la cantidad de nodos drásticamente. En el esquema planteado se aprecia una reducción de los nodos de 10000 (grilla 100x100) (figura 3) a 14 nodos (figura 4).

Se consigue además, relacionar el mapa con puntos característicos, que representan zonas de interés. Esto puede utilizarse para descripción de objetivos con un nivel mayor de abstracción, tales como puertas, pasillos, habitaciones aledañas y no coordinadas en el plano. Obteniendo además un movimiento mas natural.

Por otra parte el método depende fuertemente de la posibilidad de encontrar los puntos singulares, pero en algunos casos estos no son suficientes como ejemplifica la figura 9.

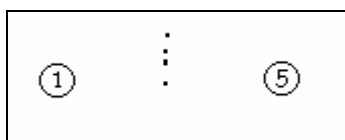


Figura 9: No hay nodos intermedios entre 1 y 5

Para solucionar este inconveniente se rodea al mapa con una serie de cargas, que permiten resolver el problema. Existen dos opciones para realizar esta tarea: cuando el entorno del robot es pequeño con respecto al alcance de los sensores, se puede definir un área de acción (rectangular o no) y bordear a esta área con cargas (figura 10a); por el contrario, cuando el entorno del robot se desconoce o es de mayor tamaño, las cargas son agregadas automáticamente y se pondrán de tal forma que describan un círculo con centro en el robot, radio igual al alcance de los sensores y equidistantes entre ellas a una distancia mayor que el ancho del robot para abarcar la posibilidad que la meta esté fuera de dicho círculo (figura 10b).

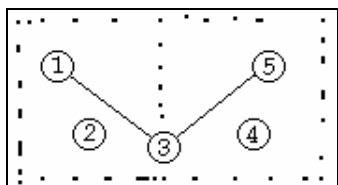


Figura 10a: Cargas adicionales para delimitar el mapa (método I para entornos reducidos)

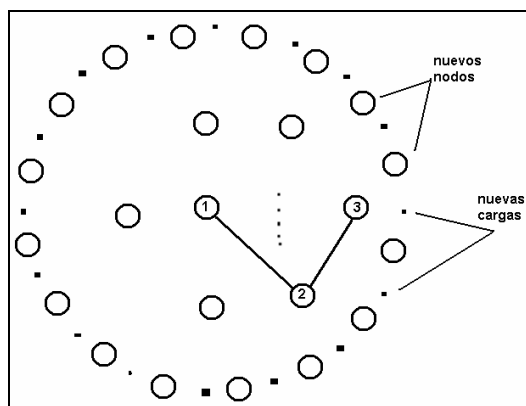


Figura 10b: Cargas adicionales para delimitar el mapa (método II para entornos desconocidos)

En el caso de la exploración de un entorno desconocido se puede delimitar el mapa explorado dibujando cargas en las zonas no exploradas. En todos los casos propuestos, la respuesta del algoritmo fue muy satisfactoria y los tiempos de procesamiento de las rutas posibles fueron disminuidos notablemente, a la vez que se pudo “visualizar” o “entender” el mapa recogido por los

sensores de otro punto de vista: no como una sucesión de celdas sino como regiones de interés.

#### 4. DISCUSION

El uso del campo eléctrico como método directo de hallar el camino fue descartado, la resolución que requería era tan alta, como para dejar de considerar las cargas puntuales y tener que pensar en barras cargadas. Esto último complicaba demasiado el proceso y el costo computacional. Además el robot puede quedar atrapado en un mínimo de potencial, necesitando un método auxiliar para destrabarlo de tal situación.

Como mecanismo alternativo, se bordean las cargas puntuales con círculos, que en caso que el campo eléctrico tratara de filtrarse por dos cargas, chocaría con estos círculos, y se lo obligaría a continuar por ellos en sentido del movimiento natural que deberían realizar si fueran barras cargadas; quedando muy similar al algoritmo Bug.

Aún se encuentra en discusión si el robot debería tener carga positiva, esto trae consigo la aparición de nuevos mínimos entre el robot y las paredes que lo rodean.

#### 5. FUTUROS TRABAJOS

El proceso de segmentación, determina la cantidad y posición de los nodos, muy importantes para la aplicación del método. Se estudiarán distintas alternativas de clasificación de puntos de interés del tipo K-means.

#### 6 BIBLIOGRAFÍA

- [1] Navarro D., Hernando Ríos, L., Parra, H.; “Sensores de Ultrasonido usados en Robótica Móvil para la medición de Distancias”, *Scientia et Técnica Año X, No 25*, Agosto 2004
- [2] Sciallato A. E., Colón D. L., Balbuena J. E.; “Técnica de Navegación Híbrida para la navegación de Robots Móviles”, *Universidad Nacional de Comahue, Neuquén, Patagonia Argentina*.
- [3] Zemansky S.; Física Volumen 2, pp. 669-695, 1998
- [4] S. Russell, P. Norving; “Inteligencia Artificial” 2da edición, pp. 110-115, 2004
- [5] M. Villafañe, A. Carimatto; “El Sonar”, *Inteligencia Artificial y Robótica IEEE UTN*, Mayo 2008.
- [6] M. I. Ribeiro; “Obstacle Avoidance”, *Institute for Systems and Robotics*, Octubre 2005
- [7] “Generación de Trayectorias robustas mediante algoritmos genéticos”, <http://descargas.cervantesvirtual.com/servlet/SirveOb>

[ras/12039418628925940987435/002760\\_11.pdf](ras/12039418628925940987435/002760_11.pdf),  
Abril 2008

[8] Grupo visión, Robótica y Proy. “Navegación  
planificada de un Robot Móvil en Entornos Interiores  
Desconocidos”, *Universidad de Murcia*